RM-77-24

THE DYNAMIC SIMPLEX METHOD*

A.I. Propoi**
V.E. Krivonozhko***

June 1977

* This research has been carried out at the Institute for Systems Studies of the State Committee for Science and Technology and the USSR Academy of Sciences in support of research on Dynamic Linear Programming carried out at IIASA.

** Institute for Systems Studies, Moscow, USSR and IIASA.

*** Institute for Systems Studies, Moscow, USSR

PREFACE

Finding optimal solutions to models is a central tool of the Systems and Decision Sciences Area, and many optimization problems in economic, management, technological systems, etc. can be reduced to dynamic linear problems. There are many different approaches and methods for tackling dynamic linear programming problems which use decomposition, penalty functions, augmented Lagrangian nested decomposition, generalized gradient, etc. methods. The simplex-method is by all means the basis method for solution of linear programming problems. However, the extension of the simplex method for the dynamic case has yet to be made.

This paper presents a finite-step algorithm which seems to be the natural and straightforward extension of the simplex-method to the dynamic case. The paper contains only a theoretical description and evaluation of the algorithm. Theoretical reasonings show that this algorithm may serve as a base for developing effective computer codes for the solution of dynamic LP problems (just as the simplex-method was for the solution of static LP problems). However, the final judgment of the algorithms' effiency can be made only after a definite period of its use in practice.

ABSTRACT

In this paper a finite-step method for solving dynamic linear programming (DLP) problems is described.

Many optimization problems in economic, management, technology, etc. are formulated as DLP problems, because now it becomes difficult to make a decision without taking into account the possible consequences of such a decision for a certain time period.

As DLP problems are large-scale by nature, the standard "static" LP methods become ineffective for the dynamic case and the development of methods specially oriented to DLP problems is needed.

The method suggested is a natural and straightforward extension of one of the most effective static LP methods--the simplex method--for DLP. A new concept--a set of local bases--(for each time step) is introduced, thus enabling considerable reduction of the requirements to computer core memory and CPU time.

In the proposed method the system of T local $m \times m$ bases is introduced and the basic simplex procedures (selection of vectors to be removed from and to be introduced into the basis, pricing procedure, transformation of bases) as applied to this system of T local bases are described. Evaluation of possibilities of the method and its connection with compact inverse LP method are discussed.

CONTENTS

The Dynamic Simplex-Method

INTRODUCTION

Methods of linear programming (LP) are now well studied and have an extensive field of applications [1,2,3]. Dynamic linear programming (DLP) is a new development of LP methods for planning and control of complex systems.

Many optimization problems in economic, management, technological systems can be reduced to DLP problems (see, for example, [1-6]). However, the development of DLP methods and its applications are restrained by lack of universal DLP computer codes. Therefore many DLP problems are now being solved by reducing them to static ones and using for their solution the standard LP codes (see, for examples, [4,6]).

As DLP problems are principally large-scale, this "static" approach is limited in its possibilities, and development of algorithms specially oriented to dynamic LP problems is needed.

In recent years, methods for DLP have been developed which make it possible to take into account the specific features of dynamic problems [7,9].* But extension of the most effective LP finite-step method--the simplex method--for the dynamic case has yet to be made.

The dynamic simplex method was suggested in [10,11]. This approach uses essentially the dynamic specific of DLP problems. The main concept of the static simplex method--the basis--is replaced by the set of local bases, introduced for the whole planning period. It allows a significant saving in the amount of computation and computer core and permits development of a set of finite-step DLP methods (primal, dual and primal-dual dynamic simplex methods) which are direct extensions of the corresponding static finite-step methods.

---

* See also references in [3].

In this paper, detailed description of the dynamic simplex method is given and connection with the method of compact inverse is discussed.

1. STATEMENT OF THE PROBLEM

Consider the DLP problem in the following canonical form.

Problem 1.1 Find a control

$$u = \{u(0),\dots,u(T-1)\}$$

and a trajectory

$$x = \{x(0),\dots,x(T)\} \quad ,$$

satisfying the state equations

$$x(t+1) = A(t)x(t) + B(t)u(t) \qquad (1.1)$$
$$(t = 0,1,\dots,T-1)$$

with initial condition

$$x(0) = x^{o} \qquad (1.2)$$

and constraints

$$G(t)x(t) + D(t)u(t) = f(t) \qquad (1.3)$$

$$u(t) \geq 0 \qquad (1.4)$$

which maximize the performance index

$$J_1(u) = a(T)x(T) \quad . \qquad (1.5)$$

Here the vector $x(t) = \{x_1(t),\dots,x_n(t)\}$ defines the state of the system at stage t in the state space X, which is assumed to be the n-dimension euclidean space; the vector $u(t) = \{u_1(t),\dots,u_r(t)\} \varepsilon E^r$ (r-dimension euclidean space) specifies the controlling action at stage t; vectors $a(T)$, x^o, $f(t)$ and the matrices $A(t)$, $B(t)$, $G(t)$, $D(t)$, are respectively of dimensions $(n \times 1)$, $(n \times 1)$, $(m \times 1)$, and $(n \times n)$, $(n \times r)$, $(m \times n)$, $(m \times r)$, and are assumed to be given.

In vector products the right vector is a column, the left vector is a row; thus, ab is the inner product of vectors a and b; aA is the product of a row-vector a on a matrix A; Aa is the product of a matrix A on a column-vector a.

The choice of a canonical form for the problem is to some extent arbitrary, various modifications and particular cases of Problem 1 being possible (i.e. integers n, m and r may depend on the number of stage t; constraints on the state and control variables can be separate; state equations include time lags of state and/or control variables; the performance index depends on the whole sequences $\{u(t)\}$ and/or $\{x(t)\}$, etc. [3,12]. However, these variants of Problem 1 can either be reduced to this problem [12,13] or the results stated below may be used directly for their solution.

Along with the primary Problem 1.1, statement of the dual problem will be necessary.

Introducing the Lagrange multipliers $\lambda(t) \in E^m$ $(t = T-1, \dots, 1, 0)$ and $p(t) \in E^n$ $(t=T,\dots,0)$ for constraints (1.3) and (1.1), (1.2) respectively. From (Table 1) one can obtain the following dual DLP problem [12].

Problem 1.2 Find a dual control

$$\lambda = \{\lambda(T-1), \dots, (0)\}$$

and a dual (conjugate) trajectory

$$p = \{p(T), \dots, p(0)\} ,$$

satisfying the costate (conjugate) equations

$$p(t) = p(t+1)A(t) - \lambda(t)G(t) \qquad (t = T-1,\dots,1,0) \tag{1.6}$$

with boundary condition

$$p(T) = a(T) \tag{1.7}$$

and constraints

$$p(t+1)B(t) - \lambda(t)D(t) \leq 0 \qquad (t = T-1,\ldots,1,0) \tag{1.8}$$

which minimize the performance index

$$J_2(\lambda) = p(0)x^o + \sum_{t=0}^{T-1} \lambda(t)f(t) \quad . \tag{1.9}$$

<u>*Definition 1.1*</u> A *feasible control* of the DLP Problem 1.1 is a vector sequence $u = \{u(0),\ldots,u(T-1)\}$ which satisfies with some trajectory $x = \{x(0),\ldots,x(T)\}$ conditions (1.1) to (1.4).

An *optimal control* of Problem 1.1 is a feasible control u^*, which maximizes (1.5).

Feasible dual controls λ and *optimal dual control* λ^* to the dual Problem 1.2 are defined in a similar way.

The sets of all feasible controls u and λ of Problems 1.1 and 2.1 will be denoted by Ω and Λ.

<u>*Theorem 1.1*</u> *(Duality Theorem* [12]*) If one of the dual Problems 1.1 and 1.2 has an optimal control, then the other has an optimal control as well and the values of the performance indexes of the primary and dual Problems 1.1 and 1.2 are equal:*

$$J_1(u^*) = J_2(\lambda^*) \quad .$$

If the performance index of either Problem 1.1 or 1.2 is unbounded (for Problem 1.1 from above and for Problem 1.2 from below), then the other problem has no feasible control.

2. AUXILIARY PROBLEM

Let $U = E^{rT}$; $u = \{u(0),\dots,u(T-1)\} \varepsilon U$ be the control space of Problem 1.1. In the control space U Problem 1.1 can be rewritten as follows.

One can obtain from the state equations (1.1) that [13]:

$$x(t) = \Psi(t,0)x(0) + \sum_{\tau=0}^{t-1} \Psi(t,\tau+1)B(\tau)u(\tau) \qquad (2.1)$$

where

$$\Psi(t,\tau) = A(t-1)\dots A(\tau) \qquad (0 \le \tau \le t),\ ,$$

$$\Psi(t,t) = I \ ,$$

I is the identity matrix.

By substituting (2.1) into (1.3) and taking into account (1.2), we obtain the constraints on controls u, given in explicit form (Table 2):

$$\begin{aligned}
&W(0,0)u(0) &&= h(0)\\
&- &&\\
&W(t,0)u(0) + \dots + W(t,t)u(t) &&= h(t)\\
&- &&\\
&W(T-1,0)u(0) + \dots + W(T-1,t)u(t) + \dots + &&\\
&\qquad + W(T-1,T-1)u(T-1) &&= h(T-1)
\end{aligned} \qquad (2.2)$$

Here

$$W(t,\tau) = G(t)\Psi(t,\tau+1)B(\tau) \qquad (t > \tau)$$

$$W(t,t) = D(t)$$

$$h(t) = f(t) - G(t)\Psi(t,0)x^0$$

The matrices $W(t,\tau)$ are of dimension $(m \times r)$ and vectors $h(t)$ are of dimension $(m \times 1)$.

The performance index (1.5) will be rewritten, respectively, in the form

$$J_1(u) = \sum_{t=0}^{T-1} q(t+1)B(t)u(t) + q(0)x^0 \tag{2.3}$$

or

$$J_1(u) = \sum_{t=0}^{T-1} c(t)u(t) + q(0)x^0 \quad ,$$

where

$$c^T(t) = q(t+1)B(t) \qquad (t = 0,\dots,T-1) \tag{2.4}$$

Here vectors q(t) are satisfied to the state equation of the form

$$q(t) = q(t+1)A(t) \qquad (t = T-1,\dots,0) \tag{2.5}$$

$$q(T) = a(T)$$

with

$$q(t) = a(T)\Psi(T,t) \quad .$$

Denoting the constraint matrix of (2.2) by W (dimension is $mT \times rT$), we can reformulate Problem 1.1 in the following equivalent form (see also Table 2):

Problem 2.1 Find a control $u = \{u(0),\dots,u(T-1)\}$, satisfying the constraints

$$\begin{aligned} Wu &= h \\ u &\geq 0 \end{aligned} \tag{2.6}$$

which maximizes the performance index

$$\tilde{J}_1(u) = cu \quad . \tag{2.7}$$

Here $h = [h(0),\dots,h(T-1)]^T$; $q = [q(0),\dots,q(T-1)]^T$; $c = [c(0),\dots,c(T-1)]^T$; T denotes transposition; $\tilde{J}_1$ differs from J_1 on the constant number $q(0)x^0$.

It is evident that the sets of optimal controls for Problems 1.1 and 2.1 are the same.

Problem 2.1 has block-triangular constraint matrix (2.2) and has been studied in many works (see [1]). However, statement of the problem constraints in the form (2.2) does not allow use of the dynamic nature of the problem in full measure. More natural, and therefore more effective, would be direct use of the specific character of Problem 1.1 as an optimal control problem.

Before considering this approach the general scheme of the simplex method as applied to Problem 2.1 will be described.

If the structure of the matrix W is not taken into account, Problem 2.1 is an ordinary LP problem in canonical form.

Let u be a feasible control; we shall define the index sets

$$I\big(u(t)\big) = \{i \mid u_i(t) > 0 , \quad i = 1,\dots,r\}$$
$$\bar{I}\big(u(t)\big) = \{i \mid u_i(t) = 0 , \quad i = 1,\dots,r\}$$

The unions of these sets over all $t = 0,1,\dots,T-1$ will be denoted by

$$I(u) = \bigcup_t I\big(u(t)\big) \; ; \qquad \bar{I}(u) = \bigcup_t \bar{I}\big(u(t)\big)$$

Denote also the columns of matrix W by $w_i(t)$ $(i = 1,\dots,r;$ $t = 0,1,\dots,T-1)$; $w_i(t) \in E^{mT}$. In that case the constraints (2.6) can be rewritten as

$$\sum_{t=0}^{T-1} \sum_{i=1}^{r} w_i(t)u_i(t) = h \; ; \qquad u_i(t) \geq 0$$

Definition 2.1 A *basic feasible control* of Problem 1.1 is a feasible control u, for which vectors $w_i(t)$, $(i,t) \in I(u)$, are linearly independent.

A *nondegenerate basic feasible control* is a basic feasible control u, for which vectors $w_i(t)$, $(i,t) \in I(u)$, constitute a basis in E^{mT}.

Evidently the basic control is an extreme point of polyhedral set Ω.

Definition 2.2 The *basis of basic control* u is a system of mT linear independent vectors $w_i(t)$, which contains all vectors $w_i(t)$, $i(t) \in I(u)$.

Denote by $I_B(u)$ the set of indices corresponding to the basic vectors $w_i(t)$; $I_N(u)$ is the set of indices corresponding to the remaining vectors $w_i(t)$ of matrix W.

In general, $I_B(u) \supseteq I(u)$. If u is a nondegenerate basic control, then

$$I_B(u) = I(u) \quad .$$

Let

$$u_B = \{u_i(t) \mid (i,t) \in I_B(u)\} \quad ,$$
$$u_N = \{u_i(t) \mid (i,t) \in I_N(u)\} \quad , \qquad (2.8)$$

and m(t) is the number of basic components of a basic control u at step t. Evidently,

$$\sum_{t=0}^{T-1} m(t) = mT \quad .$$

We shall now consider the simplex-procedure of finding the optimal control u* in terms of Problem 2.1.

As usual, without any loss in generality we assume that Problem 2.1 (1.1) is feasible and that any basic feasible control is nondegenerate.

In accordance with definitions 2.1 and 2.2 any basic feasible control may be represented as

$$u = \{u_B, u_N\}, \quad \text{with} \quad u_B \geq 0, \quad u_N = 0 \quad .$$

Let

$$u^0 = \{u_B^0, 0\} \ , \qquad u_B^0 = \{u_i^0(t)\} \ , \qquad (i,t) \in I_B(u^0)$$

be a given feasible control with associated set linearly independent vectors $w_i(t)$, $(i,t) \in I_B(u^0)$. Then

$$\sum_{(i,\tau)\in I_B(u^0)} w_i(\tau) u_i^0(\tau) = h \quad , \tag{2.9}$$

where all $u_i^0(\tau) > 0$.

Denote by W_B the matrix with columns $w_i(t)$, $(i,t) \in I_B(u^0)$ (basic matrix).

Then (2.9) can be written in the form

$$W_B \ u_B^0 = h \quad .$$

By Definition 2.2 W_B is a nonsingular matrix, therefore

$$u_B^0 = W_B^{-1} h \quad . \tag{2.10}$$

Define also the number z_0, associated with the basic control $u^0 = \{u_B^0, 0\}$:

$$z_0 = \sum_{(i,\tau)\varepsilon I_B(u^0)} c_i(\tau)u_i^0(\tau) \quad , \tag{2.11}$$

Or, in vector form,

$$z_0 = c_B u_B^0 \quad ,$$

where $c_B = \{c_i(\tau)\}$, $(i,\tau) \varepsilon I_B(u^0)$.

Here $c_i(\tau)$ are the cost coefficients of the objective function (2.7) and defined in (2.4), z_0 is the corresponding value of the objective function for the given control u^0.

Since the set of vectors $w_i(t)$, $(i,t) \varepsilon I_B(u^0)$, which constitutes the basic matrix W_B, is linearly independent, we can express any column vector of the matrix W in terms of vectors $\{w_i(t)\}$, $(i,t) \varepsilon I_B(u^0)$.

Let $w_j(t_1)$ be an arbitrary column vector of W ($j = 1,\ldots,r$; $t_1 = 0,1,\ldots,T-1$), then

$$w_j(t_1) = \sum_{(i,\tau)\varepsilon I_B(u^0)} v_{ij}(t_1,\tau)w_i(\tau) \tag{2.12}$$

($j = 1,\ldots,r$; $t_1 = 0,1,\ldots,T-1$).

Or, in matrix form,

$$w_j(t_1) = W_B v_j(t_1) \quad , \tag{2.12a}$$

where $v_j(t_1) = \{v_{ij}(t_1,\tau)\}$, $(i,\tau) \varepsilon I_B(u^0)$, ($j = 1,\ldots,r$; $t_1 = 0,1,\ldots,T-1$) are coefficients vectors of dimension mT and defined from

$$v_j(t_1) = W_B^{-1} w_j(t_1) \quad . \tag{2.13}$$

Using the coefficients $\{v_{ij}(t_1,\tau)\}$, one can define numbers

$$z_j(t_1) = \sum_{(i,\tau)\varepsilon I_B(u^0)} v_{ij}(t_1,\tau)c_i(\tau) \tag{2.14}$$

where $c_i(\tau)$ are calculated from (2.4), or in the matrix form:

$$z_j(t_1) = v_j(t_1)c_B \quad .$$

Thus, numbers $c_j(t)$ and $z_j(t)$ can be defined from

$$\begin{aligned} c_j(t) &= \sum_{k=1}^{n} q_k(t+1)b_{kj}(t) \\ z_j(t) &= \sum_{(i,\tau)\varepsilon I_B(u^0)} \sum_{k=1}^{n} q_k(t+1)b_{ki}(\tau)v_{ij}(t,\tau) \end{aligned} \tag{2.15}$$

$$(j = 1,\ldots,r; \quad t = 0,1,\ldots,T-1) \quad ,$$

or, in the matrix form:

$$c_j(t) = q(t+1)b_j(t) \tag{2.15a}$$

$$z_j(t) = \sum_{\tau=0}^{T-1} q(\tau+1)B_B(\tau)v_j(t,\tau)$$

Here $b_{kj}(t)$ are elements of the matrix $B(t)$, $b_j(t)$ are its columns; the matrix $B_B(\tau)$ is generated by the basic columns $b_i(\tau),(i,\tau)\varepsilon I_B(u^0)$ of the matrix $B(\tau)$.

The following assertions are true (see, for example, [1,14]).

<u>*Theorem 2.1*</u> *If for any basic feasible control* $u^0 = \{u_B^0, 0\}$ *the conditions*

$$z_j(t) - c_j(t) \geq 0$$

hold for all $j = 1,\ldots,r$ *and* $t = 0,1,\ldots,T-1$, *then* u^0 *is an optimal control.*

If for any fixed (j,t) *the condition*

$$z_j(t) - c_j(t) < 0$$

holds, then a set of feasible controls can be constructed, such that

$$\tilde{J}_1(u) \quad > \quad z_0 \quad = \quad \tilde{J}_1(u^0)$$

for any control u *of the feasible control set* Ω*, where the upper bound of* $\tilde{J}_1(u)$ *is either finite or infinite.*

Here $\tilde{J}_1(u)$ and z_0 are defined by (2.7) and (2.11) respectively.

Thus the direct implementation of the simplex method to Problem 1.1 (2.1) gives the following procedure. It is assumed that an initial basic feasible control and, associated with it, numbers $v_{ij}(t,\tau)$, $c_j(t)$, $z_j(t)$ $(i,\tau) \in I_B(u)$, $j = 1,\ldots,r$, $t = 0,1,\ldots,T-1$ (initial tableau) has been constructed:

1. The testings of the $z_j(t) - c_j(t)$, $(j = 1,\ldots,r;\ t = 0,\ldots,T-1)$ elements to determine whether an optimal control has been found, that is, whether $z_j(t) - c_j(t) \geq 0$ for all j and t.

   If $z_j(t) - c_j(t) < 0$ for some (j,t) and all $v_{ij}(t,\tau)$ associated with the pair (j,t) are non-positive, then the Problem 1.1 (2.1) has no solution, that is, the function $\tilde{J}_1(u)$ is unbounded above on the feasible control set Ω.

   If $z_j(t) - c_j(t) < 0$ for some pair $(j,t) \in I_N(u)$, and for each such pair (j,t) at least one $v_{ij}(t,\tau) > 0$, then a new basic control is chosen. For that one should proceed to new steps.

2. The selection of the vector to be introduced into the basis, that is selection of the vector with minimal value of $z_j(t) - c_j(t)$.

   Let the pair of indices associated with this vector be (j,t_1).

3. The selection of the vector to be eliminated from the basis.

4. The transition from the old basic feasible control to the new one. The new basic feasible control $u^{(1)} = \{u_B^{(1)}, 0\}$ is defined by

$$\begin{aligned} u_i^{(1)}(\tau) &= u_i(\tau) - \theta_0 v_{ij}(t_1,\tau), & (i,\tau) \in I_B(u^0) \\ u_j^{(1)}(t_1) &= \theta_0 & \\ u_i^{(1)}(\tau) &= 0 \qquad (i,\tau) \neq (j,t_1); & (i,\tau) \in I_N(u^0) \quad , \end{aligned}$$

where the value θ_0 is calculated from

$$\theta_0 = \min_{\substack{v_{ij}(t_1,\tau) > 0 \\ (i,\tau) \in I_B(u^0)}} \frac{u_i(\tau)}{v_{ij}(t_1,\tau)} = \frac{u_\ell(t_2)}{v_{\ell j}(t_1,t_2)} \qquad (2.16)$$

For the nondegenerate case the minimum in (2.16) achieves for a single pair (ℓ, t_2). For the degenerate case the minimum achieves for several pairs (i,τ) and it is necessary to use special rule for choice (i,τ) in order to avoid zigzagging of the algorithm [1,14].

In any case after a finite number of steps the process terminates, as either an optimal control will be found or insolubility of the problem will be established.

In practice the numbers $z_j(t)$ are usually computed from

$$z_j(t) = \sum_{(i,\tau) \in I_B(u^0)} w_{ij}(t) \lambda_i(\tau) \quad , \qquad (2.17)$$

where $\lambda = \{\lambda_i(\tau),\ (i,\tau) \in I_B(u^0)\}$ are simplex multipliers for the basis W_B associated with the basic feasible control u^0:

$$\lambda = c_B W_B^{-1} \quad . \qquad (2.18)$$

The general scheme considered above is in practice ineffective for solution of Problem 1.1 (2.1) because of the too-large dimension of the matrix W. Further, the input data are usually given in the form of Problem 1.1 rather than in the form of Problem 2.1. Therefore the simplex procedure directly designed for solution of Problem 1.1 will be described.

3. EQUIVALENT PROBLEM

The matrices $D(t)$ $(t=0,\dots,T-1)$ of constraints (1.3) will be assumed to have the rank m. This preposition, as in the static case, is not limiting [1,14].

Let us denote

$$\hat{f}(0) = f(0) - G(0)x(0) \quad .$$

Then constraints (1.3) can be rewritten as

$$D(0)u(0) = \hat{f}(0) \quad . \tag{3.1}$$

In accordance with the preposition we can choose m linear independent column-vectors $d_i(0)$ of the matrix $D(0)$ and generate the matrix $D_0(0)$ from these columns. The matrix from the rest of the columns will be denoted by $D_1(0)$. Thus

$$D(0) = [D_0(0);\ D_1(0)] \quad .$$

As determinant $|D_0(0)| \neq 0$, the constraints (3.1) can be rewritten in the form

$$u_0(0) = D_0^{-1}(0)\hat{f}(0) - D_0^{-1}(0)D_1(0)u_1(0) \quad , \tag{3.2}$$

where components of the vector $u_0(0) \in E^m$ correspond to the matrix $D_0(0)$ and components of the vector $u_1(0) \in E^{r-m}$ correspond to the matrix $D_1(0)$.

To representation of the matrix

$$D(0) \;=\; [D_0(0);\; D_1(0)]$$

corresponds the representation of the matrix

$$B(0) \;=\; [B_0(0);\; B_1(0)] \quad .$$

Therefore

$$x(1) \;=\; A(0)x(0) + B_0(0)u_0(0) + B_1(0)u_1(0) \quad . \qquad (3.3)$$

Substituting (3.2) into (3.3), we obtain

$$x(1) \;=\; x^*(1) + B^1(0)u_1(0) \quad , \qquad (3.4)$$

where

$$B^1(0) \;=\; B_1(0) - B_0(0)D_0^{-1}(0)D_1(0) \quad ,$$

$$x^*(1) \;=\; A(0)x(0) + B_0(0)u_0^*(0) \quad ,$$

$$u_0^*(0) \;=\; D_0^{-1}(0)\hat{f}(0) \quad .$$

Considering the constraints (1.2) of Problem 1.1 at the next step and inserting the value x(1) defined by (3.4) into (1.2), we obtain the constraints at the next step in the following form:

$$G(1)x^*(1) + G(1)B^1(0)u_1(0) + D(1)u(1) \;=\; f(1) \qquad (3.5)$$

Denote

$$\hat{D}(1) \;=\; [G(1)B^1(0);\; D(1)] \quad , \qquad (3.6)$$

$$\hat{u}(1) \;=\; [u_1(0),\; u(1)]^T \quad , \qquad (3.7)$$

$$\hat{f}(1) \;=\; f(1) - G(1)x^*(1) \quad . \qquad (3.8)$$

Now (3.5) is rewritten as

$$\hat{D}(1)\hat{u}(1) \;=\; \hat{f}(1) \quad , \qquad (3.9)$$

where the matrix $\hat{D}(1)$ is of dimension $(m \times (2r-m))$, and vectors $\hat{u}(1) \in E^{2r-m}$; $\hat{f}(1) \in E^m$ are defined from (3.6) to (3.8).

As the structure of constraints (3.9) is identical to (3.1), then the construction for the zero step $t = 0$ can be repeated for the next step, that is for (3.9).

As a result, we obtain at a step t, $0 \leq t \leq T-1$, the following relations.

Let

$$\hat{D}(t)\hat{u}(t) = \hat{f}(t) \tag{3.10}$$

where

$$\hat{D}(t) = [G(t)B^1(t-1);\ D(t)] \tag{3.11}$$

$$\hat{u}(t) = [\hat{u}_1(t-1);\ u(t)]^T \tag{3.12}$$

$$\hat{f}(t) = f(t) - G(t)x^*(t) \quad . \tag{3.13}$$

In (3.11) to (3.13) the matrix $B^1(t-1)$ and vectors $\hat{u}_1(t-1), x^*(t)$ are defined from recurrent relations, which will be defined below, at the transfer from step t to step t+1.

By construction, the matrix $\hat{D}(t)$ includes m linearly independent columns $\hat{d}_i(t)$.

Definition 3.1 The set of m linearly independent columns $\hat{d}_i(t)$ of the matrix $\hat{D}(t)$ is called the *local basis* at the step t ($t = 0,1,\ldots,T-1$).

The matrix formed from these columns will be denoted by $\hat{D}_0(t)$; the matrix formed from the remaining columns -- by $\hat{D}_1(t)$. Thus, (3.10) can be rewritten as

$$\hat{D}_0(t)\hat{u}_0(t) + \hat{D}_1(t)\hat{u}_1(t) = \hat{f}(t) \tag{3.14}$$

$$\hat{D}(t) = [\hat{D}_0(t),\ \hat{D}_1(t)] \quad . \tag{3.14}$$

Hence

$$\hat{u}_0(t) = \hat{D}_0^{-1}(t)\hat{f}(t) - \hat{D}_0^{-1}(t)\hat{D}_1(t)\hat{u}_1(t) \quad . \tag{3.15}$$

Or

$$\hat{u}_0(t) = \hat{u}_0^*(t) - \Phi(t)\hat{u}_1(t) \quad , \tag{3.16}$$

where

$$\hat{u}_0^*(t) = \hat{D}_0^{-1}(t)\hat{f}(t) \quad , \tag{3.17}$$

$$\Phi(t) = \hat{D}_0^{-1}(t)\hat{D}_1(t) \quad . \tag{3.18}$$

Let

$$x(t) = x^*(t) + B^1(t-1)\hat{u}_1(t-1) \quad , \tag{3.19}$$

where $x^*(t)$ and $B^1(t-1)$ will be defined later.

By substituting (3.19) into state equation (1.1), we obtain

$$x(t+1) = A(t)x^*(t) + \hat{B}(t)\hat{u}(t) \quad , \tag{3.20}$$

where

$$\hat{B}(t) = [A(t)B^1(t-1); B(t)] \quad , \tag{3.21}$$

the vector $\hat{u}(t)$ is defined by (3.12).

Considering the representation

$$\hat{B}(t) = [\hat{B}_0(t); \hat{B}_1(t)] \quad , \tag{3.22}$$

$$\hat{u}(t) = [\hat{u}_0(t); \hat{u}_1(t)]^T \tag{3.23}$$

and substituting (3.16) into (3.19), we obtain again (cf. (3.19)):

$$x(t+1) = x^*(t+1) + B^1(t)\hat{u}_1(t) \quad , \tag{3.24}$$

where

$$x^*(t+1) = A(t)x^*(t) + \hat{B}_0(t)\hat{u}_0^*(t) \quad , \tag{3.25}$$

$$B^1(t) = \hat{B}_1(t) - \hat{B}_0(t)\Phi(t) \quad . \tag{3.26}$$

Initial conditions for (3.24), (3.25), (3.10) are

$$\begin{aligned} x^*(0) &= x(0) \\ \hat{B}(0) &= B(0) \\ \hat{D}(0) &= D(0) \quad . \end{aligned} \tag{3.27}$$

The specific of representation (3.23), (3.24), (3.10) is a recurrent determination of control $\hat{u}(t)$, that is, using (3.12), we obtain

$$\begin{aligned} \hat{u}(t) &= [\hat{u}_1(t-1), u(t)]^T = \\ &= [\hat{u}_2(t-2), u_1(t-1), u(t)]^T = \dots \\ &= [u_t(0), u_{t-1}(1), \dots, u_{t-i}(i), \dots, u_1(t-1), u(t)]^T \end{aligned} \tag{3.28}$$

where the vector $u_{t-i}(i)$ is formed from those components of the control u which are recomputed from a step i to the step t by virtue of the procedure which was described above. The relations (3.28) show that the vector u(t) may include components $u_i(\tau)$ from preceding steps $\tau = t-1,\dots,1,0$.

Consider now the last step

$$\hat{D}_0(T-1)\hat{u}_0(T-1) + \hat{D}_1(T-1)\hat{u}_1(T-1) = \hat{f}(T-1) \tag{3.29}$$

where $\hat{D}_0(T-1)$ is a nonsingular matrix.

Let

$$\hat{u}_1(T-1) = 0 \quad . \tag{3.30}$$

Then from (3.29):

$$\hat{u}_0(T-1) = \hat{D}_0^{-1}(T-1)\hat{f}(T-1) \quad . \tag{3.31}$$

Determining the value of the vector $\hat{u}(T-1) = [\hat{u}_0(T-1), \hat{u}_1(T-1)]^T$ from (3.30), (3.31), one can determine the values of feasible control $\{u(t)\}$ for a given set of local bases $\{D_0(t)\}$ $(t = 0,1,\dots,T-1)$.

This procedure will be called Procedure 1 (Table 3).

Procedure 1 reduces the original Problem 1.1 to an equivalent Problem 3.1.

Problem 3.1 (Equivalent Problem) Find a control $u = \{u(t)\}$, for which

$$c(T)\hat{B}_0(T-1)\hat{u}_0(T-1) \rightarrow \max$$

$$\hat{u}_0(t) = \hat{u}_0^*(t) - \Phi(t)\hat{u}_1(t) \geq 0$$

$$\hat{u}_0^*(t) = \hat{D}_0^{-1}(t)[f(t) - G(t)x^*(t)]$$

$$x^*(t-1) = A(t)x^*(t) + \hat{B}_0(t)u_0^*(t)$$

$$\hat{u}(t) = [\hat{u}_1(t-1), u(t)] = [\hat{u}_0(t); \hat{u}_1(t)]$$

$$\hat{D}(t) = [\hat{D}_0(t); \hat{D}_1(t)]$$

$$\hat{B}(t) = [\hat{B}_0(t); \hat{B}_1(t)]$$

$$\Phi(t) = \hat{D}_0^{-1}(t)\hat{D}_1(t)$$

$$\hat{u}(T-1) = \hat{u}_0(T-1) \; ; \quad \hat{u}_1(T-1) = 0$$

$$\hat{D}(T-1) = \hat{D}_0(T-1) \; ; \quad \hat{D}_1(T-1) = 0$$

$$\hat{B}(T-1) = \hat{B}_0(T-1) \; ; \quad \hat{B}_1(T-1) = 0$$

$$t = 0, 1, \ldots, T-1 .$$

Procedure 1 gives the values of both vectors $u(t)$ and vectors $\{\hat{u}_0(t), \hat{u}_1(t)\}$ $(t = 0, 1, \ldots, T-1)$.

The set of all indices (i,t) associated with the components of vectors $\hat{u}_0(t)$ will be denoted by $I_0(u)$; the supplement of $I_0(u)$ to the total set of indices $\{(i,t) \mid i = 1, \ldots, r; \; t = 0, 1, \ldots, T-1\}$ will be denoted by $\bar{I}_0(u)$. One can easily see that the total number of indices of $I_0(u)$ is equal to mT and that the total number of indices $\bar{I}_0(u)$ is equal to $(r-m)T$.

Theorem 3.1 *Let a control* u *be computed from Procedure 1 for a given set of local bases* $\{D_0(t)\}$ *with boundary conditions*

$$\hat{u}_0(T-1) = \hat{D}^{-1}(T-1)\hat{f}(T-1)$$
$$\hat{u}_1(T-1) = 0$$

and let

$$u_i(t) \geq 0 \quad \textit{for all} \quad (i,t) \in I_0(u) \quad .$$

Then u *is a basic feasible control and*

$$u = \{u_B, u_N\} \quad ;$$
$$u_B = \{u_i(t) \mid (i,t) \in I_0(u)\} \quad ;$$
$$u_N = \{u_i(t) \mid (i,t) \in I_N(u)\} \quad .$$

Proof Let W be the matrix which is generated by the columns $w_i(t)$ of the constraint matrix W, associated with variables $\hat{u}_0(t)$, that is,

$$W_0 = \| w_i(t) \| , \quad (i,t) \in I_0(u) \quad .$$

By construction, W_0 is a square matrix of dimension $mT \times mT$.

For proof of the theorem we shall need the following assertion.

Lemma 3.1 *The matrix* W_0 *is nonsingular if and only if the matrices* $\hat{D}_0(t)$ $(t = 0,1,\ldots,T-1)$ *are nonsingular.*

Proof: Sufficiency The procedure of computing $\{\hat{u}_0(t)\}$ described above is a block modification of the Gauss method [15] where pivot blocks are matrices $\hat{D}_0(t)$. The Gauss algorithm transforms the matrix W_0 to an upper-block triangular matrix with $\hat{D}_0(t)$ on its diagonal:

$$W_0 = \begin{bmatrix} \hat{D}_0(0)x & 0 & \cdots & & 0 \\ 0 & \hat{D}_0(1)x & \cdots & & 0 \\ \cdots & & & & \cdots \\ 0 & \cdots & \hat{D}_0(t)x & \cdots & 0 \\ \cdots & & & & \cdots \\ 0 & \cdots & & & \hat{D}_0(T-1) \end{bmatrix}$$

The Gauss algorithm does not change the rank of the original matrix [15]. In fact, the relation

$$\left| |W_0| \right| = \left| |D_0(0)| \ . \ . \ . \ |D_0(T-1)| \right| \tag{3.32}$$

holds, where $\left| |W| \right|$ is the absolute value of the determinant of a matrix W. The relation (3.32) implies that, if matrices $\hat{D}_0(t)$ $(t = 0,1,...,T-1)$ are nonsingular, then the matrix W_0 is also nonsingular.

Necessity Suppose that k iterations of the Gauss algorithm have been done and W_o^k is a matrix obtained after k iterations:

$$W_0^k = \begin{bmatrix} \hat{D}_0(0)x & \cdots & & & 0 \\ 0 & \hat{D}_0(1)x & \cdots & & 0 \\ \cdots & & & & \cdots \\ \cdots & & D_0(k-1)x & \cdots & \cdots \\ \cdots & & & & \cdots \\ 0 & 0 & \cdots & & \tilde{W}_0^k \end{bmatrix}$$

Here $\tilde{W}_0^k$ is a submatrix, generated by elements of W_0^k which are outside of pivot rows and columns of previous iterations. In this case the relation (3.32) should be replaced by

$$\left| |W_0| \right| = \left| |\hat{D}_0(0)| \ . \ . \ . \ |\hat{D}_0(k-1)| |\tilde{W}_0^k| \right| \ .$$

The first block-row of $\tilde{W}_0^k$ is $[\hat{D}(k);0]$. Suppose that the matrix $\hat{D}(k)$ cannot generate any nonsingular square submatrix $\hat{D}_0(k)$ of dimension m. This implies that the rows of the matrix $\hat{D}(k)$ are linearly dependent and the matrix $\tilde{W}_0^k$ is singular with $|\tilde{W}_0^k| = 0$. Then it follows, from (3.32), that $|W_0| = 0$, which contradicts the assumption of the lemma.

Thus, if the matrix W_0 is nonsingular then at each step of the Gauss algorithm a nonsingular matrix $\hat{D}_0(k)$ can be constructed. This completes the proof of the lemma.

The proof of the theorem now follows directly. By definition, matrices $\hat{D}_0(t)$ $(t = 0,\ldots,T-1)$ are nonsingular, which implies that the matrix W_0 is also nonsingular and vectors $w_i(t)$, $(i,t) \in I_0(u)$, are linear independent.

It follows from Procedure 1 that

$$u_i(t) = 0 \quad \text{for all} \quad (i,t) \in \bar{I}_0(u) \quad .$$

As $u_i(t) \geq 0$ for all $(i,t) \in I_0(u)$, then in accordance with Definition 2.1 u is a basic feasible control. This completes the proof of the theorem.

Thus Procedure 1 provides the possibility in principle of computing a basic feasible control, operating only with the set of local bases $\{\hat{D}_0(t)\}$.

It follows from the constructions of the previous section that a basic feasible control $u = \{u_B;u_N\}$ is a solution of the set of linear equations

$$W_B u_B = h \tag{3.33}$$

with

$$u_B \geq 0 \ , \quad u_N = 0 \quad .$$

The proof of Theorem 3.1 shows that implementation of the Gauss algorithm to solution of the set of equations (3.33) permits operation not with the inverse W_B^{-1} of dimension $mT \times mT$

but with T inverses $\hat{D}_0^{-1}(t)$ of dimension $m \times m$.

Hence the computing possibilities enlarge in qualitative degree. Thus Procedure 1 is basic to this approach. However, as will be seen further, it is not used in this precise form.

4. CONTROL VARIATION

Let

$$u = \{u_B;\ u_N\}$$

be a basic feasible control of Problem 1.1 with a basic matrix (basis) W_B.

In accordance with Theorem 3.1 the basis W_B is equivalent to the set of local bases $\{\hat{D}_{0B}(t)\}$. Therefore our problem is to develop the simplex operations for solution of Problem 1.1 relative to the set of local bases $\{\hat{D}_{0B}(t)\}$. (This operation has been described in Section 2 in connection with the "global" basis W_B.)

For a given basic feasible control u, let us fix the pair of indices (j,t_1) $(j = 1,\dots,r;\ t_1 = 0,1,\dots,T-1)$ such that the corresponding column $d_j(t_1)$ of the matrix $D(t_1)$ is not in the basis, that is,

$$(j,t_1) \in I_N(u) \quad .$$

We first consider the procedure for selection of the column $d_j(t_1)$ to be introduced into the basis, that is, into the set of local bases $\{\hat{D}_{0B}(t)\}$. In accordance with Section 3, the constraints (1.3) at step t can be written as

$$\hat{D}_{0B}(t)\hat{u}_{0B}(t) + \hat{D}_{1B}(t)\hat{u}_{1B}(t) = \hat{f}(t) \tag{4.1}$$

where

$$[\hat{D}_{0B}(t);\ \hat{D}_{1B}(t)] = \hat{D}_B(t) \quad ;$$

$$[\hat{u}_{0B}(t);\ \hat{u}_{1B}(t)] = \hat{u}_B(t) \quad ;$$

$$\hat{u}_B(t) \geq 0 \quad .$$

Here the subscript B denotes submatrices and/or vectors associated with a given basis $W_B \leftrightarrow \{\hat{D}_{0B}(t)\}$.

Let a vector $\hat{v}^*_{0B}(t_1) \in E^m$ define representation of the vector $d_j(t_1)$ in terms of column-vectors of the matrix $D_{0B}(t_1)$, that is,

$$\hat{D}_{0B}(t_1)\hat{v}^*_{0B}(t_1) = d_j(t_1) \quad . \tag{4.2}$$

As $\hat{D}_{0B}(t_1)$ is a nondegenerate matrix, one can obtain from (4.2)

$$\hat{v}^*_{0B}(t_1) = \hat{D}^{-1}_{0B}(t_1)d_j(t_1) \quad . \tag{4.3}$$

Taking into account (4.2) and (4.3), we can rewrite (4.1) as

$$\hat{D}_{0B}(t_1)\left[\hat{u}_{0B}(t_1) - \theta\hat{v}^*_{0B}(t_1)\right] + \\ + \hat{D}_{1B}(t_1)\hat{u}_{1B}(t_1) + \theta d_j(t_1) = \hat{f}(t_1) \tag{4.4}$$

where θ is a real number.

It is evident that the equality (4.4) is true for any value of the parameter θ.

It follows from (4.4) that a new control $\hat{u}^\theta(t_1)$ is introduced at step t_1:

$$\hat{u}^\theta(t_1) = \left[\hat{u}^\theta_{0B}(t_1);\ \hat{u}^\theta_{1B}(t_1);\ \hat{u}^\theta_N(t_1)\right]^T \quad ,$$

where

$$\begin{aligned} \hat{u}^\theta_{0B}(t_1) &= \hat{u}_{0B}(t_1) - \theta\hat{v}^*_{0B}(t_1) \\ \hat{u}^\theta_{1B}(t_1) &= \hat{u}_{1B}(t_1) \\ \hat{u}^\theta_N(t_1) &= [0, \ldots, \theta, \ldots, 0]^T \quad . \end{aligned} \tag{4.5}$$

By substituting the control $\hat{u}^{\theta}(t_1)$ in state equation (3.20), we obtain

$$x^{\theta}(t_1+1) = A(t_1)x^*(t_1) + \hat{B}(t_1)\hat{u}^{\theta}(t_1) \quad .$$

Or

$$x^{\theta}(t_1+1) = x(t_1+1) - \theta y^*(t_1+1) \quad , \qquad (4.6)$$

where

$$x(t_1+1) = x^*(t_1+1) + B_B^1(t_1)\hat{u}_{1B}(t_1) \quad ;$$

$$y^*(t_1+1) = \hat{B}_{0B}(t_1)\hat{v}^*_{0B}(t_1) - b_j(t_1) \quad ; \qquad (4.7)$$

$$\hat{B}(t_1) = \left[A(t_1)B_B^1(t_1-1);\ B(t_1)\right] \quad ;$$

$b_j(t_1)$ is the j-th column of the matrix $B(t_1)$.

Substituting (4.6) into constraints (1.3), we see that they will be true if

$$\hat{D}_B(t_1+1)\hat{u}_B^{\theta}(t_1+1) - \theta G(t_1+1)y^*(t_1+1) = \hat{f}(t_1+1). \quad (4.8)$$

Let us express the vector $-G(t_1+1)y^*(t_1+1)$ in terms of column vectors of the matrix $\hat{D}_{0B}(t_1+1)$:

$$\hat{D}_{0B}(t_1+1)\hat{v}^*_{0B}(t_1+1) = -G(t_1+1)y^*(t_1+1) \quad , \qquad (4.9)$$

where

$$\hat{v}^*_{0B}(t_1+1) = -\hat{D}_{0B}^{-1}(t_1+1)G(t_1+1)y^*(t_1+1) \quad . \qquad (4.10)$$

Considering (4.9),(4.10), the equality (4.8) can be rewritten as

$$\hat{D}_{0B}(t_1+1)\left[\hat{u}_{0B}(t_1+1) - \theta\hat{v}^*_{0B}(t_1+1)\right] +$$
$$+ \hat{D}_{1B}(t_1+1)\hat{u}_{1B}(t_1+1) + \theta G(t_1+1)y^*(t_1+1) = \hat{f}(t_1+1)$$

where

$$\left[\hat{D}_{0B}(t_1+1);\hat{D}_{1B}(t_1+1)\right] = \hat{D}_B(t_1+1) \quad .$$

We see that the introduction of the compensating term into the equality (4.8) is equivalent to the introduction of a new control $\hat{u}^{\theta}(t_1+1)$ at step t_1+1:

$$\hat{u}^{\theta}(t_1+1) = [\hat{u}^{\theta}_{0B}(t_1+1);\ \hat{u}^{\theta}_{1B}(t_1+1);\ \hat{u}^{\theta}_{N}(t_1+1)] \quad ;$$

where

$$\begin{aligned} \hat{u}^{\theta}_{0B}(t_1+1) &= \hat{u}_{0B}(t_1+1) - \theta\hat{v}^{*}_{0B}(t_1+1) \\ \hat{u}^{\theta}_{1B}(t_1+1) &= \hat{u}_{1B}(t_1+1) \\ \hat{u}^{\theta}_{N}(t_1+1) &= 0 \quad . \end{aligned} \tag{4.11}$$

Thus the variation of the control (4.5) at step t_1, where vector $\hat{v}^{*}_{0B}(t_1)$ is defined by (4.3), induces a variation of control (4.11) at the next steps $\tau = t_1+1,\ t_1+2,\ \ldots,\ T-2$ with

$$\hat{v}^{*}_{0B}(t) = -\hat{D}^{-1}_{0B}(\tau)G(\tau)y^{*}(\tau) \quad . \tag{4.12}$$

Vectors $y^{*}(\tau)$ are satisfied to the following difference equation:

$$y^{*}(\tau+1) = A(\tau)y^{*}(\tau) + \hat{B}_{0B}(\tau)\hat{v}^{*}_{0B}(\tau) \tag{4.13}$$

where vectors $\hat{v}^{*}_{0B}(\tau)$ $(\tau = t_1+1,\ldots,T-1)$ are defined from (4.12) and vector $\hat{v}^{*}_{0B}(t_1)$ is defined from (4.3).

Now we consider the last step:

$$\hat{D}_B(T-1)\left[\hat{u}_B(T-1) - \theta\hat{v}_B(T-1)\right] -$$
$$- \theta G(T-1)y^{*}(T-1) = \hat{f}(T-1) \quad . \tag{4.14}$$

As $u = \{u_B, 0\}$ is a basic feasible control, then by virtue of Theorem 3.1 the matrix $\hat{D}_B(T-1)$ is nonsingular and

$$\hat{D}_B(T-1) = \hat{D}_{0B}(T-1) \quad .$$

Therefore (4.14) yields that

$$\hat{v}_B(T-1) = \hat{v}^*_{0B}(T-1) = -\hat{D}^{-1}_{0B}(T-1)G(T-1)y^*(T-1) \quad (4.15)$$

By definition, the structure of vector $\hat{v}_B(T-1)$ is similar to the structure of vector $\hat{u}_B(T-1)$. Hence, define a vector:

$$\hat{v}_B(T-1) = \left[\hat{v}_{1B}(T-2),\ v_B(T-1)\right] \quad (4.16)$$

where vector $v_B(T-1)$ is associated with the variation of vector $u_B(T-1)$, vector $\hat{v}_{1B}(T-2)$ is associated with the variation of vector $\hat{u}_{1B}(T-2)$:

$$\hat{u}^{\theta}_{1B}(T-2) = \hat{u}_{1B}(T-2) - \theta\hat{v}_{1B}(T-2) \quad .$$

To satisfy the constraints at step $T-2$ the additional term $-\theta\hat{D}_{1B}(T-2)v_{1B}(T-2)$ must be compensated by the additional variation $\hat{v}^1_{0B}(T-2)$ of control $\hat{u}_{0B}(T-2)$:

$$\hat{u}^{\theta}_{0B}(T-2) = \hat{u}_{0B}(T-2) - \theta\left[\hat{v}^*_{0B}(T-2) - \hat{v}^1_{0B}(T-2)\right] \quad ,$$

where

$$\hat{v}^1_{0B}(T-2) = \hat{D}^{-1}_{0B}(T-2)\hat{D}_{1B}(T-2)\hat{v}_{1B}(T-2) =$$

$$= \Phi_B(T-2)\hat{v}_{1B}(T-2) \quad .$$

Let

$$\hat{v}_{0B}(T-2) = \hat{v}^*_{0B}(T-2) - \hat{v}^1_{0B}(T-2) \quad .$$

As in the case of (3.12) and (3.23), we can write

$$\hat{v}_B(T-2) = \left[\hat{v}_{0B}(T-2), \hat{v}_{1B}(T-2)\right] =$$

$$= \left[\hat{v}_{1B}(T-3),\ \hat{v}_B(T-2)\right] \quad . \quad (4.17)$$

By induction we find that in order to satisfy the constraints (1.2) for all θ and $\tau = 0,1,\ldots,T-1$, we must define

$$\hat{D}_B(T-1)[\hat{u}_B(T-1) - \theta\hat{v}_B(T-1)] - \theta G(T-1)y^*(T-1) = \hat{f}(T-1)$$

if $\tau = T-1$;

$$\hat{D}_{0B}(\tau)[\hat{u}_{0B}(\tau) - \theta(\hat{v}^*_{0B}(\tau) - \hat{v}^1_{0B}(\tau))] + \hat{D}_{1B}(\tau)[\hat{u}_{1B}(\tau) - \theta\hat{v}_{1B}(\tau)] - \theta G(\tau)y^*(\tau) = \hat{f}(\tau) \quad (4.18)$$

if $t_1 + 1 \leq \tau \leq T-2$;

$$\hat{D}_{0B}(t_1)[\hat{u}_{0B}(t_1) - \theta(\hat{v}^*_{0B}(t_1) - \hat{v}^1_{0B}(t_1))] + \hat{D}_{1B}(t_1)[\hat{u}_{1B}(t_1) - \theta\hat{v}_{1B}(t_1)] + \theta d_j(t_1) = \hat{f}(t_1) \quad (4.19)$$

if $\tau = t_1$;

$$\hat{D}_{0B}(\tau)[\hat{u}_{0B}(\tau) + \theta\hat{v}^1_{0B}(\tau)] + \hat{D}_{1B}(\tau)[\hat{u}_{1B}(\tau) - \theta\hat{v}_{1B}(\tau)] = \hat{f}(\tau) \quad (4.20)$$

if $0 \leq \tau \leq t_1 - 1$.

The vectors $\hat{v}^*_{0B}(\tau)$ must satisfy the following relations:

$$\hat{v}^*_{0B}(T-1) = -\hat{D}^{-1}_{0B}(T-1)G(T-1)y^*(T-1) = \hat{v}_B(T-1) ,$$

if $\tau = T-1$;

$$\hat{v}^*_{0B}(\tau) = -\hat{D}^{-1}_{0B}(\tau)G(\tau)y^*(\tau)$$

if $t_1 + 1 \leq \tau \leq T-2$;

$$\hat{v}^*_{0B}(t_1) = \hat{D}^{-1}_{0B}(t_1)d_j(t_1)$$

if $\tau = t_1$.

The vectors $\hat{v}^1_{0B}(\tau)$ satisfy the relations $(0 \leq \tau \leq T-2)$:

$$\hat{v}^1_{0B}(\tau) = \hat{D}^{-1}_{0B}(\tau)\hat{D}_{1B}(\tau)\hat{v}_{1B}(\tau) = \Phi_B(\tau)\hat{v}_{1B}(\tau) \qquad (4.21)$$

Thus the variation $\hat{v}_{0B}(\tau)$ of control $\hat{u}_{0B}(\tau)$ $(\tau = 0,1,\ldots, T-1)$ is defined by:

$$\begin{aligned} \hat{v}_{0B}(T-1) &= \hat{v}^*_{0B}(T-1) \quad ; \\ \hat{v}_{0B}(\tau) &= \hat{v}^*_{0B}(\tau) - \hat{v}^1_{0B}(\tau) \; , \quad \text{if } t_1 \leq \tau \leq T-2 \qquad (4.22) \\ \hat{v}_{0B}(\tau) &= -\hat{v}^1_{0B}(\tau) \; , \quad \text{if } 0 \leq \tau \leq t_1 \quad . \end{aligned}$$

Using (4.16), (4.17) we can hence define the values of vectors $\{v_B(\tau)\}$ associated with the variation of control $\{u_B(\tau)\}$. Thus, if a new column $w_j(t_1)$ associated with a column $d_j(t_1)$ is introduced into the basis W_B, then the variation of a basic feasible control $\{u_B, u_N\}$ in the origin control space U is defined by

$$u^\theta_i(\tau) = \begin{cases} u_i(\tau) - \theta v_{iB}(\tau) \; , & \text{if } (i,\tau) \in I_B(u) \\ \theta \; , & \text{if } (i,\tau) = (j,t_1) \\ 0 \; , & \text{if } (i,\tau) \in I_N(u) \; , \\ & \quad (i,\tau) \neq (j,t_1) \quad . \end{cases} \qquad (4.23)$$

The upper row in (4.23) can also be rewritten as

$$\hat{u}^\theta_{0B}(\tau) = \hat{u}_{0B}(\tau) - \theta\hat{v}_{0B}(\tau) \quad . \qquad (4.24)$$

Let us summarize the results.

The variation of a basic feasible control u is defined by relations (4.23); the sequence of vectors v(t) is computed from the following procedure (Table 4).

First the sequence of vectors $\hat{v}^*_{0B}(\tau)$ $(\tau = t_1, t_1+1, \ldots, T-1)$ is defined by recurrent equations (4.13), (4.12) with initial conditions (4.3).

The sequence of vectors $\hat{v}^1_{0B}(\tau)$ $(\tau = T-2, \ldots, 1, 0)$ is computed from recurrent relations

$$\hat{v}_B(\tau) = \left[\hat{v}_{1B}(\tau - 1, \hat{v}_B(\tau)\right] ,$$

$$\hat{v}_B(\tau) = \left[\hat{v}_{0B}(\tau), \hat{v}_{1B}(\tau)\right] ,$$

with boundary condition (4.15), where vectors $\hat{v}_{0B}(\tau)$ are defined by (4.22) and (4.21).

We shall refer to the determining of the variation $\{u^\theta(\tau)\}$ of a feasible control $\{u(t)\}$ as Procedure 2. Procedure 2 is represented schematically in Figure 1 and Table 4.

Thus,

$$\hat{u}^\theta_{0B} = \hat{u}_{0B} - \theta\hat{v}_{0B}$$

and sequences $\hat{u}^\theta_{0B}$, $\hat{u}_{0B}$ and $\hat{v}_{0B}$ are associated with sequences $\{\hat{u}^\theta(\tau)\}$, $\{\hat{u}(\tau)\}$ and $\{\hat{v}(\tau)\}$ respectively.

The variation $\{\hat{u}^\theta(\tau)\}$ of the basic feasible control $\{\hat{u}(\tau)\}$ is satisfied to the constraints (1.1) to (1.3) of Problem 1.1 by definition.

As $\{\hat{u}(\tau)\}$ is a feasible control, then the constraints (1.4) will also be satisfied for sufficiently small $\theta \geq 0$. Hence the control $\{\hat{u}^\theta(\tau)\}$ is feasible if $0 \leq \theta \leq \theta_0$. The value of θ_0 is defined by relations (cf. (2.6)):

$$\theta_0 = \min_{(i,\tau)} \frac{\hat{u}_{0i}(\tau)}{\hat{v}_{0i}(\tau)} = \frac{\hat{u}_{0\ell}(t_2)}{\hat{u}_{0\ell}(t_2)} \qquad (4.25)$$

$$\hat{v}_{0i}(\tau) > 0$$

where $\hat{u}_{0i}(\tau)$, $\hat{v}_{0i}(\tau)$ are the i-th components of vectors $\hat{u}_{0B}(\tau)$, $\hat{v}_{0B}(\tau)$.

The equality (4.25) follows from (1.4) and (4.24); minimum in (4.25) achieves at single pair (ℓ, t_2) in the nondegenerate case.

Let us now define the variation of trajectory $\{x(t)\}$. Considering (3.23), (4.6), (4.16), (4.18) to (4.20), we find that the variation of trajectory

$$x^{\theta}(\tau) = x(\tau) - \theta y(\tau) \qquad (\tau = 1,\ldots,T)$$

will be defined by

$$\begin{aligned} y(T) &= y^*(T) \\ y(\tau+1) &= y^*(\tau+1) + B_B^1(\tau)\hat{v}_{1B}(\tau) \qquad \tau = T-2,\ldots,1,0 \end{aligned} \qquad (4.26)$$

where the vectors

$$y^*(\tau) = 0$$

if $0 \le \tau \le t_1$,

and

$$y^*(\tau+1) = A(\tau)y^*(\tau) + \hat{B}_{0B}(\tau)\hat{v}^*_{0B}(\tau)$$

if $t_1+1 \le \tau \le T-1$.

5. OBJECTIVE FUNCTION VARIATION

The special feasible variation of a basic feasible control has been built up in the previous section. Now we determine the corresponding variation of the performance index (objective function) (1.5) under this variation of control.

Let

$$d_j(t_1) \ , \qquad (j,t_1) \in I_N(u)$$

be a column vector to be introduced to the basis W_B.

In accordance with (4.26),

$$J_1(u^\theta) = a(T)x(T) - \theta a(T)y^*(T) \quad .$$

Denote the variation of the objective function by

$$\Delta_j(t_1) \equiv \Delta J_1(u^\theta) = J_1(u^\theta) - J_1(u) = a(T)y^*(T) \ , \qquad (5.1)$$

where indices (j,t_1) show that the variation has been caused by introduction of the column $d_j(t_1), (j,t_1) \in I_N(u)$ to the basis.

By substituting $y^*(T)$ from

$$y^*(T) = A(T-1)y^*(T-1) + \hat{B}_{0B}(T-1)\hat{v}^*_{0B}(T-1)$$

into (5.1), we obtain

$$\Delta_j(t_1) = a(T)A(T-1)y^*(T-1) + $$
$$+ a(T)\hat{B}_{0B}(T-1)\hat{v}^*_{0B}(T-1) \quad . \qquad (5.2)$$

Considering (4.16), (3.20) and (2.3), rewrite (5.2) as

$$\Delta_j(t_1) = q(T-1)y^*(T-1) +$$
$$+ q(T-1)B^1_B(T-2)\hat{v}_{1B}(T-2) +$$
$$+ q(T)B_B(T-1)v_B(T-1) \quad , \qquad (5.3)$$

where $B_B(T-1)$ is the matrix generated by basis columns of the matrix $B(T-1)$, variation $v_B(T-1)$ is associated with basic components of the vector $u_B(T-1)$.

By substituting

$$y^*(T-1) = A(T-2)y^*(T-2) + \hat{B}_{0B}(T-2)\hat{v}^*_{0B}(T-2)$$

into (5.3) and again using (2.3), we obtain

$$\begin{aligned}\Delta_j(t_1) = \; & q(T-2)y^*(T-2) + \\ & + q(T-1)\hat{B}_{0B}(T-2)\hat{v}^*_{0B}(T-2) + \\ & + q(T-1)B^1_B(T-2)\hat{v}_{1B}(T-2) + \\ & + q(T)B_B(T-1)v_B(T-1) \quad . \end{aligned} \tag{5.4}$$

Considering (3.26) and (4.21), (4.22), we can express $\Delta_j(t_1)$ in the form

$$\begin{aligned}\Delta_j(t_1) = \; & q(T-2)y^*(T-2) + \\ & + q(T-1)\hat{B}_{0B}(T-2)\hat{v}_{0B}(T-2) + \\ & + q(T-1)\hat{B}_{1B}(T-2)\hat{v}_{1B}(t_1-2) + \\ & + q(T)B_B(T-1)v_B(T-1) \quad . \end{aligned}$$

Hence and from (3.21) it follows that

$$\begin{aligned}\Delta_j(t_1) = \; & q(T-2)y^*(T-2) + \\ & + q(T-1)\hat{B}_B(T-2)\hat{v}_B(T-2) + \\ & + q(T)B_B(T-1)v_B(T-1) \quad . \end{aligned}$$

Eventually by induction we obtain for all $(j,t_1) \in I_N(u)$:

$$\Delta_j(t_1) = \sum_{\tau=0}^{T-1} q(\tau+1)B_B(\tau)v_B(\tau) - q(t_1+1)b_j(t_1) \tag{5.5}$$

One can see that vectors $v_B(\tau)$ $(\tau = 0,1,\ldots,T-1)$ are a solution of the equations system (2.12). The solution is obtained by means of the compact inverse matrix Procedure 2, which is analogous to Procedure 1 of basic feasible control computation (see Tables 3 and 4).

In this procedure we used notations (cf. (2.12),(2.13) and notations of section 4):

$$v_{ij}(t_1,\tau) = v_{iB}(\tau) \quad , \qquad (j,t_1) \in I_N(u)$$

where

$$v_{iB}(\tau) = v_B(\tau) \quad , \qquad (i,\tau) \in I_B(u) \quad .$$

Thus formulas (5.5) and (2.15a) coincide and we can write

$$\Delta_j(t_1) = z_j(t_1) - c_j(t_1) =$$

$$= \sum_{\tau=0}^{T-1} q(\tau+1)B_B(\tau)v_B(\tau) - q(t_1+1)b_j(t_1) \qquad (5.6)$$

Using the dual Problem 1.2, we can now obtain another form for the definition of the objective function variation $\Delta_j(t_1)$. This form corresponds to (2.17) and is more convenient in practice.

By substituting the expression $\hat{v}^*_{0B}(T-1)$ from (4.12) at $\tau = T-1$ into (5.2), one can obtain

$$\Delta_j(t_1) = a(T)A(T-1)y^*(T-1) -$$

$$- a(T)\hat{B}_{0B}(T-1)\hat{D}_{0B}^{-1}(T-1)G(T-1)y^*(T-1) \qquad (5.7)$$

Define a vector $\lambda(T-1)$ as

$$\lambda(T-1) = a(T)\hat{B}_{0B}(T-1)\hat{D}_{0B}^{-1}(T-1) \quad . \qquad (5.8)$$

Then

$$\Delta_j(t_1) = p(T-1)y^*(T-1) \quad , \qquad (5.9)$$

where the vector $p(T-1)$ is computed from dual state equation (1.6) with boundary condition (1.7) at $t = T-1$, that is,

$$p(T-1) = a(T)A(T-1) - \lambda(T-1)G(T-1) \quad .$$

By induction we obtain

$$z_j(t_1) = \lambda(t_1)d_j(t_1) \qquad (5.10)$$

$$c_j(t_1) = p(t_1+1)b_j(t_1) \qquad (5.11)$$

$$\Delta_j(t_1) = z_j(t_1) - c_j(t_1) \qquad (5.12)$$

$$(j,t_1) \in I_N(u) \quad ,$$

where

$$\lambda(t) = p(t+1)\hat{B}_{0B}(t)\hat{D}_{0B}^{-1}(t) \qquad (5.13)$$

and the variables $\lambda(t)$,$p(t+1)$ are satisfied to the dual state equation (1.6) with boundary condition (1.7).

Theorem 5.1 *Vectors* $\{\lambda(t)\}$ *computed from (5.13), (1.6) and (1.7) are the simplex-multipliers for the basis* W_B.

Proof It is sufficient to show, in accordance with the definition of simplex-multipliers [1,2,12,14], that vectors $\lambda(t)$ are satisfied to the dual constraints (1.8) as equalities for basic indices; that is,

$$p(t+1)b_j(t) - \lambda(t)d_j(t) = 0 \quad , \quad (j,t) \in I_B(u) \quad .$$

For this, let us consider the constraints (1.8) of the dual Problem 2.1 relative to the current basis W_B of the primal Problem 1.1. They can be written at $t=0$ as

$$\lambda(0)D_B(0) = p(1)B_B(0) \quad . \qquad (5.14)$$

As a nonsingular matrix $\hat{D}_{0B}(0)$ can be generated by columns of the matrix $D_B(0)$, then (5.14) can be rewritten as

$$\lambda(0)\hat{D}_{0B}(0) = p(1)\hat{B}_{0B}(0) \qquad (5.15)$$

$$\lambda(0)\hat{D}_{1B}(0) = p(1)\hat{B}_{1B}(0) \quad . \qquad (5.16)$$

We use here the constructions of section 3.

From (5.15), (5.16) we obtain

$$p(1)\left[B_{0B}(0)\hat{D}_{0B}^{-1}(0)\hat{D}_{1B}(0) - \hat{B}_{0B}(0)\right] = 0 \qquad (5.17)$$

or, in accordance with (3.25),

$$p(1)B_B^1(0) = 0 \quad . \qquad (5.18)$$

Using the state equations (1.6), the condition (5.18) can be rewritten as

$$p(2)A(1)B_B^1(0) - \lambda(1)G(1)B_B^1(0) = 0 \quad . \qquad (5.19)$$

Hence and from (1.8) we obtain, for the next step,

$$\lambda(1) = p(2)\hat{B}_{0B}(1)\hat{D}_{0B}^{-1}(0) \quad . \qquad (5.20)$$

By induction,

$$\lambda(t) = p(t+1)\hat{B}_{0B}(t)\hat{D}_{0B}^{-1}(t) \qquad (5.21)$$

holds for all $t = 1,2,\ldots,T-1$, where matrices $\hat{B}_{0B}(t)$ and $\hat{D}_{0B}^{-1}(t)$ are defined in section 3. This completes the proof.

Define Procedure 3 by formulas (5.13), (1.6), (1.7). Procedure 3 allows computation of the values of simplex multipliers $\{\lambda(t)\}$ for the current basis W_B (Table 5).

Both Procedure 1 for finding a primal basic feasible control and Procedure 3 for finding the corresponding dual control (simplex multipliers) $\{\lambda(t)\}$ are based on the generalized Gauss algorithm. It should be noted that for computing both the values of vectors $\{\lambda(t),p(t+1)\}$ and the values of vectors $\{u(t),x(t)\}$, one can use the same matrices $\hat{D}_{0B}^{-1}(t)$, $\hat{D}_{1B}(t)$, $\hat{B}_{0B}(t)$, $B_B^1(t)$.

6. TRANSFORMATION OF THE BASIS

The procedure of computing the values

$$\Delta_j(t) \quad = \quad z_j(t) - c_j(t)$$

for vectors $d_j(t)$, $(j,t) \in I_N(u)$, which are not in the basis allows us, in accordance with (4.24), to define the vector to be introduced into the basis and the vector to be removed from the basis.

Let a column vector

$$d_j(t_1)$$

be introduced into the basis, and a column vector

$$\hat{d}_{0\ell}(t_2)$$

be removed from the basis.

Here $d_j(t_1)$ is the j-th nonbasic column of the matrix $D(t_1)$ and $\hat{d}_{0\ell}(t_2)$ is the ℓ-th column of the matrix $\hat{D}_{0B}(t_2)$, $0 \leq t_1, t_2 \leq T-1$.

Replacing the vector $d_j(t_1)$ by the vector $\hat{d}_{0\ell}(t_2)$ implies the transformation of the old system of local bases $\{\hat{D}_{0B}(t)\}$ into a new system of local bases $\{D'_{0B}(t)\}$.

As in the case of the static simplex-method, this procedure is one of the crucial ones which essentially defines the efficiency of the algorithm.

In the revised simplex-method the inverse basis matrix is transformed by multiplying on an elementary matrix. The advantage in generating the inverse for each basis by means of elementary matrices is that only a minimal amount of information need be stored.

In the dynamic simplex-method we operate with the system of inverses $\{\hat{D}_{0B}^{-1}(t) \ (t=0,1,\ldots,T-1)\}$ of local bases. Hence the efficiency of the dynamic simplex-method will be directly defined by the scheme of transformation of inverses $\{\hat{D}_{0B}^{-1}(t)\}$.

The difficulty of building such a transformation is determined by the fact that, first, the transformation of a local basis at step t changes the subsequent local bases $\hat{D}_{0B}(\tau)$ $(\tau = t+1,\dots,T-1)$ and, second, the vector $\hat{d}_{0\ell}(t_2)$, which should be removed from the basis, may belong to the local basis $\hat{D}_{0B}(t_2)$ at another step t_2, $t_2 \neq t_1$.

The theorem given below defines the sufficient condition when the replacement of a basis column in a local basis $\hat{D}_{0B}(t)$ does not change the other local bases.

<u>*Theorem 6.1*</u> *The replacement of the* i*-th column in a local basis* $\hat{D}_{0B}(t)$ *does not change the other local bases, if the* i*-th row of matrices*

$$\Phi_B(t) = \hat{D}_{0B}^{-1}(t)\hat{D}_{1B}(t) \tag{6.1}$$

vanishes.

<u>*Proof*</u> When we replace the i-th column in the matrix $\hat{D}_{0B}(t)$, then in accordance with (6.1), the transformation of the matrix $\Phi_B(t)$ will be similar to the transformation of the inverse $\hat{D}_{0B}^{-1}(t)$, that is, the i-th pivot row of the matrix is added to the other row with some coefficients [1,14].

Therefore, if the i-th row of the matrix $\Phi_B(t)$ is equal to zero, the matrix $\Phi_B(t)$ will not change. In accordance with (3.26) the matrix $B_B^1(t)$ does not change either. Considering (3.11),(3.20) at $\tau = t+1$, we find that all subsequent local bases $(\tau = t+1,t+2,\dots,T-1)$ also do not change.

<u>*Consequence 6.1*</u> *If an element* $\phi_{ij}(t)$ *of the matrix* $\Phi_B(t)$ *is equal to zero, then the replacement of the* i*-th column in the local basis* $\hat{D}_{0B}(t)$ *does not change the* j*-th column in the matrix* $B_B^1(t)$.

Now we describe some auxiliary operations.

Let us consider the relation (4.1) and replace the ℓ-th column of the matrix $\hat{D}_{0B}(t)$ by a column of the matrix $\hat{D}_{1B}(t)$.

It is known from the simplex-method algorithm that existence of a nonzero pivot element in the ℓ-th row of the matrix $\Phi_B(t)$ is sufficient for the matrix $\hat{D}_{0B}(t)$ to be nonsingular [1,14].

Let

$$\phi_{\ell q}(t) \neq 0$$

and the q-th component of the vector $\hat{u}_{1B}(t)$ be the q-th component of the vector $\hat{u}_{0B}(t+1)$.

When the ℓ-th column of the matrix $\hat{D}_{0B}(t)$ and the q-th column of the matrix $\hat{D}_{1B}(t)$ interchange, the inverse $\hat{D}_{0B}^{-1}(t)$ is transformed by multiplying from the left on the elementary matrix. The elementary matrix has dimension $m \times m$ and differs from the identity matrix by the ℓ-th column with components [1,14]:

$$\eta_i = -\frac{\phi_{iq}(t)}{\phi_{\ell q}(t)} \qquad i = 1,\ldots,m \quad (i \neq \ell) \quad ;$$

$$\eta_\ell = \frac{1}{\phi_{\ell q}(t)} \quad .$$

The replacement of columns in matrices $\hat{B}_{0B}(t)$ and $\hat{B}_{1B}(t)$ is carried out in a similar way. The matrix $B_B^1(t)$ is transformed as follows [16]:

$$[B_B^1(t)]^T = B_B^1(t)E_q \quad , \tag{6.2}$$

where E_q is the square elementary row matrix, which differs from the identity matrix by the q-th row with components

$$\xi_i(t) = \frac{\phi_{\ell i}(t)}{\phi_{\ell q}(t)} \quad , \qquad i \neq q \quad ;$$

$$\xi_i(t) = -\frac{1}{\phi_{\ell q}(t)} \quad , \qquad i = q \quad .$$

The order of the matrix E_q equals the number of columns of the matrix $B_B^1(t)$.

Define now the transformation of the inverses $\hat{D}_{0B}^{-1}(\tau)$ $(\tau = t+1,\dots,T-1)$. Taking into account the structure of the matrices $\hat{D}_B(t+1)$ and $\hat{B}_B(t+1)$, we can write (with accuracy up to the interchanging of columns):

$$\begin{aligned}\hat{D}_B(t+1) &= [G(t+1)B_B^1(t+1);\ D_B(t+1)] = \\ &= [\hat{D}_{0B}(t+1);\ \hat{D}_{1B}(t+1)] \end{aligned} \tag{6.3}$$

and

$$\begin{aligned}\hat{B}_B(t+1) &= [A(t+1)B_B^1(t+1);\ B_B(t+1)] = \\ &= [\hat{B}_{0B}(t+1),\ \hat{B}_{1B}(t+1)]\ . \end{aligned} \tag{6.4}$$

Considering (6.2) to (6.4), we obtain

$$\begin{bmatrix} \hat{D}'_{0B}(t+1) & \hat{D}'_{1B}(t+1) \\ \hat{B}'_{0B}(t+1) & \hat{B}'_{1B}(t+1) \end{bmatrix} = \begin{bmatrix} \hat{D}_{0B}(t+1) & \hat{D}_{1B}(t+1) \\ \hat{B}_{0B}(t+1) & \hat{B}_{1B}(t+1) \end{bmatrix} \begin{bmatrix} M_q & N_q \\ 0 & E \end{bmatrix} \tag{6.5}$$

Here $\hat{D}'_B(t+1)$, $\hat{B}'_B(t+1)$ are the matrices corresponding to a new basis; M_q is the elementary row matrix of dimension $m \times m$; the matrix N_q consists of zeros except in the q-th row; its dimension equals $m \times k$, where k is the number of columns $\hat{D}_{1B}(t+1)$; E is the identity matrix of dimension $k \times k$.

The right matrix in (6.5) is built up as follows: the matrix E_q is enlarged up to dimension $(m+k) \times (m\times k)$ in such a way that in the added part the main diagonal contains units and all the rest added elements are zero; then the elements of the q-th row are interchanged in accordance with the interchange of columns of the matrix $\hat{D}_B(t+1)$ when it generates the matrices $\hat{D}_{0B}(t+1)$ and $\hat{D}_{1B}(t+1)$.

It is shown in [16] that if the transformation (6.5) is taking place, the following relations hold:

$$
\begin{aligned}
[\hat{D}_{0B}^{-1}(t+1)]' &= M_q^{-1}\hat{D}_{0B}^{-1}(t+1) \\
\hat{B}_{0B}'(t+1) &= \hat{B}_{0B}(t+1)M_q \\
\Phi_B'(t+1) &= M_q^{-1}N_q + M_q^{-1}\Phi_B(t+1) \quad .
\end{aligned}
\tag{6.6}
$$

The matrix $B_B'(t+1)$ doesn't change, therefore all the subsequent local bases do not change either.

This procedure we shall call *the interchange of the* ℓ*-th column of the matrix* $\hat{D}_{0B}(t)$ *with the* q*-th column of the matrix* $\hat{D}_{0B}(t)$.

Now let us consider the interchange of the ℓ-th column of the matrix $\hat{D}_{0B}(t)$ with some column of the matrix $\hat{D}_{0B}(t^*)$, $t^* > t+1$.

In the ℓ row of the matrix $\Phi_B(t)$, let the first nonzero element $\phi_{\ell q}(t)$ correspond to the basic variable, which is recomputed to the local basis $\hat{D}_{0B}(t^*)$, and all elements $\phi_{\ell i}(t)$ which correspond to the variable are recomputed to local bases $\hat{D}_{0B}(\tau)$, $t < \tau < t^*$, equal to zero. Now we partition the matrices $\Phi_B(t)$ and $\hat{B}_{1B}(t)$ into two parts:

$$
\begin{aligned}
\Phi_B &= [\Phi_1(t);\ \Phi_2(t)] \quad ; \\
\hat{B}_{1B} &= [\hat{B}_{11}(t);\ \hat{B}_{12}(t)] \quad .
\end{aligned}
$$

Let the columns corresponding to the variables which are recomputed into the local bases $\hat{D}_{0B}(\tau)$, $t < \tau < t^*$, enter the matrix $\Phi_1(t)$ $(\hat{B}_{11}(t))$, and the rest columns enter the matrix $\Phi_2(t)$ $(\hat{B}_{12}(t))$.

Then in accordance with (7.2) and Consequence 7.1, the matrix $B_1^1(t)$ does not change at the interchange of the ℓ-th column of the matrix $\hat{D}_{0B}(t)$ with the q-th column of the matrix $\hat{D}_{1B}(t)$. The matrix $B_2^1(t)$, which is defined from

$$B_B^1(t) = [B_1^1(t);\ B_2^1(t)] \quad ,$$

is transformed in accordance with the formula

$$[B_2^1(t)]' = B_2^1(t)E_k \quad ,$$

where k is the number of the column of the matrix $\Phi_2(t)$ which contains the element $\phi_{\ell q}(t)$. The order of matrix E_k is equal to the number of columns of matrix $B_2^1(t)$.

Let the k-th column of matrix $\Phi_2(t)$ correspond to the k-th component of vector $\hat{u}_{0B}(t^*)$.

In accordance with the partitioning of the matrix $B_B^1(t)$ and (6.3), (6.4), the columns of matrices

$$G(t+1)B_2^1(t) \quad \text{and} \quad A(t+1)B_2^1(t)$$

do not enter the matrices

$$\hat{D}_{0B}(t+1) \quad \text{and} \quad \hat{B}_{0B}(t+1) \quad .$$

Therefore the matrices $\hat{D}_{0B}(t+1)$, $\hat{B}_{0B}(t+1)$ do not change.

Let us partition the matrices $\Phi_B(t+1)$, $B_B^1(t+1)$ and $\hat{B}_{1B}(t+1)$ into two submatrices

$$\Phi_B(t+1) = [\Phi_1(t+1);\ \Phi_2(t+1)] \quad ;$$

$$B_B^1(t+1) = [B_1^1(t+1);\ B_2^1(t+1)] \quad ;$$

$$B_{1B}^1(t+1) = [\hat{B}_{11}(t+1);\ \hat{B}_{12}(t+1)] \quad .$$

The columns of the matrix $\Phi_B(t+1)$, which correspond to the same basic elements as the columns of the matrix $\Phi_2(t)$, enter the matrix $\Phi_2(t+1)$.

In accordance with the partitioning, the matrices $\Phi_1(t+1)$ and $\hat{B}_{11}(t+1)$ do not change with interchange of the columns.

The matrices $\Phi_2(t+1)$ and $\hat{B}_{12}(t+1)$ are transformed by formulas

$$\begin{aligned} \Phi_2'(t+1) &= \Phi_2(t+1)E_k \quad , \\ \hat{B}_{12}'(t+1) &= \hat{B}_{12}(t+1)E_k \quad . \end{aligned} \tag{6.7}$$

As

$$\hat{B}_2^1(t+1) = \hat{B}_{12}(t+1) - \hat{B}_{0B}(t+1)\Phi_2(t+1)$$

then, taking into account (6.7), we obtain

$$\left[B_2^1(t+1)\right]' = B_2^1(t+1)E_k \quad . \tag{6.8}$$

Similar reasoning is valid up to the step t^*. Thus, the interchange of the q-th column of the matrix $\hat{D}_{0B}(\tau)$ with the k-th column of the matrix $\hat{D}_{0B}(t^*)$ causes changes neither in the local bases $\hat{D}_{0B}(\tau)$ nor in the matrices $\hat{B}_{0B}(\tau)$ $(\tau = t+1,\dots,t^*-1)$; the matrices $\Phi_2(\tau)$ and $B_2^1(\tau)$ are transformed by formulas (6.7), (6.8) if $t+1=\tau$ $(\tau = t+1, t+2, \dots, t^*-1)$.

At step t^* part of the columns of the matrix $G(t^*)B_2^1(t^*-1)$ enters the matrix $\hat{D}_{0B}(t^*)$. Therefore the transformation of the matrices at this step reduces to the case considered above (see (6.5),(6.6)).

This procedure we shall call *the interchange of the* ℓ*-th column of the matrix* $\hat{D}_{0B}(t)$ *with the* k*-th column of the matrix* $\hat{D}_{0B}(t^*)$*, where* $t^* > t+1$.

The procedures of interchangement of columns of the matrices $\hat{D}_{0B}(t)$ and $\hat{D}_{0B}(t^*)$ $(t^* > t+1)$ allow us to describe the transformation procedure of the old local bases $\{\hat{D}_{0B}(t)\}$ into new ones $\{\hat{D}_{0B}'(t)\}$.

When a vector $\hat{d}_{0\ell}(t_2)$ is replaced by a vector $d_j(t_1)$, two cases are possible.

Case 1: $t_2 < t_1$

In this case the ℓ-th row of the matrix $\Phi_B(t)$ contains a nonzero pivot element.

In fact, the number of the variable to be introduced into the basis is defined by the relation (4.25). Hence the ℓ-th component of the vector $\hat{v}_{0B}(t_2)$ is not zero.

From (4.21), (4.22) we find that

$$\hat{v}_{0B}(t_2) = -\Phi_B(t_2)\hat{v}_{1B}(t_2) \quad \text{if} \quad t_2 < t_1 \quad .$$

Therefore the ℓ-th row of the matrix $\Phi_B(t_2)$ contains at least one nonzero element.

Let the pivot element correspond to the j-th component of vector $\hat{u}_{0B}(t_2 + \tau)$.

Replace the ℓ-th element of the matrix $\hat{D}_{0B}(t_2)$ by the j-th element of the matrix $\hat{D}_{0B}(t_2 + \tau)$. This interchange does not change the basic solution. Therefore, if $t_2 + \tau < t_1$, the above reasonings are true and we can proceed with the interchanges.

In result we obtain the following case.

Case 2: $t_2 \geq t_1$

Proceeding with these subsequent interchanges, we remove the vector to be eliminated into such a local basis $\hat{D}_{0B}(t_3)$, $t_3 \geq t_1$, which satisfies the condition of Theorem 6.1.

If such $t_3 \leq T - 1$ does not exist, then we replace the column to be removed into the last local basis $\hat{D}_{0B}(T - 1)$.

In turn the column to be removed can be replaced in the local basis $\hat{D}_{0B}(t_3)$.

Let the vector to be removed be the ℓ-th column of the matrix $\hat{D}_{0B}(t_3)$. Before introducing the vector $d_j(t_1)$ into the basis it is necessary to recompute it at the step t_3.

In result we obtain

$$\hat{v}^*_{0B}(t_1) = \hat{D}^{-1}_{0B}(t_1)d_j(t_1) ,$$

$$y^*(t_1+1) = -b_j(t_1) + \hat{B}_{0B}(t_1)\hat{v}^*_{0B}(t_1) ,$$

$$\hat{v}^*_{0B}(\tau) = -\hat{D}^{-1}_{0B}(\tau)G(\tau)y^*(\tau) , \qquad (6.9)$$

$$y^*(\tau+1) = A(\tau)y^*(\tau) + \hat{B}_{0B}(\tau)\hat{v}^*_{0B}(\tau)$$

$$\tau = t_1+1,\ t_1+2,\ \ldots,\ t_3 .$$

In these formulas the new local bases $\{\hat{D}'_{0B}(t)\}$ obtained in the result of the above interchanges have been introduced.

The above-considered interchange of the column to be introduced is possible as the ℓ-th (pivot) element of the vector $\hat{v}^*_{0B}(t_3)$ is not equal to zero.

In fact, in accordance with (4.25) the ℓ-th element of the vector $\hat{v}^*_{0B}(t_3)$ is not zero. Transformation formulas (6.9) coincide with the formulas (5.3), (5.7), (5.12), (5.13).

In accordance with (4.21), (4.22),

$$\hat{v}_{0B}(t_3) = \hat{v}^*_{0B}(t_3) - \Phi_B(t_3)\hat{v}_{1B}(t_3) .$$

But, as the ℓ-th row of the matrix $\Phi_B(t_3)$ vanishes,

$$\hat{v}_{0\ell}(t_3) = \hat{v}^*_{0\ell}(t_3) \neq 0 .$$

Thus a new set of local bases is obtained.

7. CONNECTION WITH THE METHOD OF COMPACT INVERSE

The method considered above has another interpretation connected with the factorized representation of the inverse.

We need the following assertion.

Theorem 7.1 [15] *Let a nonsingular square matrix* F *be partitioned into blocks*

$$F = \begin{bmatrix} \overbrace{H}^{m} & \overbrace{P}^{n} \\ Q & R \end{bmatrix} \begin{matrix} \}m \\ \}n \end{matrix} ,$$

where H *is a nonsingular matrix.*

Then F *is represented in the form*

$$F = \bar{F} \cdot U = \begin{bmatrix} H & 0 \\ Q & C \end{bmatrix} \cdot \begin{bmatrix} \overbrace{I_m}^{m} & \overbrace{\phi}^{n} \\ 0 & I_n \end{bmatrix} \begin{matrix} \}m \\ \}n \end{matrix} ,$$

where

$$C = R - QH^{-1}P , \quad |C| \neq 0 , \quad \phi = H^{-1}P ,$$

I_m *and* I_n *are the identity matrices of appropriate dimensions.*

Theorem 8.1 is not stated in [15] in explicit form, but directly follows from results given in [15].

The structure of constraints of Problems 1.1 implies the following structure of its basis matrix:

$$\bar{B} = \begin{bmatrix} D_B(0) & & & & & & & \\ B_B(0) & -I & & & & & & \\ & G(1) & D_B(1) & & & & & \\ & A(1) & B_B(1) & -I & & & & \\ & & & \cdot & & & & \\ & & & & \cdot & & & \\ & & & & & \cdot & & \\ & & & & & -I & & \\ & & & & & G(T-1) & D_B(T-1) & \\ & & & & & A(T-1) & B_B(T-1) & -I \end{bmatrix} \quad (7.1)$$

where I is the identity matrix of dimension $n \times n$, $D_B(t)$ and $B_B(t)$ are submatrices, consisting of basic columns of the constraints matrices (1.1) and (1.3).

As the rows of $D_B(0)$ are linearly independent, one can choose m linearly independent columns in the matrix $D_B(0)$. These columns generate the matrix $\hat{D}_{0B}(0)$.

By column interchange we can transform the matrix $D_B(0)$ and obtain

$$D_B(0) = \left[\hat{D}_{0B}(0);\ \hat{D}_{1B}(0)\right] ,$$

where $\hat{D}_{1B}(0)$ is the submatrix, consisting of the columns of the matrix $D_B(0)$ which are not in the matrix $\hat{D}_{0B}(0)$.

The column interchange of the matrix $B_B(0)$ is fulfilled in a similar way:

$$B_B(0) = \left[\hat{B}_{0B}(0);\ B_{1B}(0)\right] .$$

In accordance with Theorem 7.1, one can show that the matrix $\bar{B}$ is expressed as

$$\bar{B} = \bar{B}_0 U_0 \quad (7.2)$$

where U_0 is the upper triangular matrix of dimension coinciding with the dimension of the matrix $\bar{B}$:

$$U_0 = \begin{bmatrix} 1 & & & & \Phi_B(0) \\ & \cdot & & & \\ & & \cdot & & \\ & & & \cdot & \\ & & & & 1 \end{bmatrix} .$$

In the matrix U_0 the dimension and location of the matrix

$$\Phi_B(0) = \hat{D}_{0B}^{-1}(0)\hat{D}_{1B}(0)$$

coincide with the dimension and location of the matrix $\hat{D}_{1B}(0)$ in $\bar{B}$, the matrix $\bar{B}_0$ is obtained from the matrix $\bar{B}$ through replacement $[\hat{D}_{0B};0]$ by $[\hat{D}_{0B}(0);0]$ and $\hat{B}_{1B}(0)$ by $B_B^1(0) = \hat{B}_{1B}(0) - \hat{B}_{0B}(0)\Phi_B(0)$.

In the matrix $\bar{B}_0$ we interchange the submatrix $-E$ and the submatrix $B_B^1(0)$; then we interchange the submatrices $G(1)$ and $A(1)$ in the matrix $\bar{B}_0$ and the submatrix $\Phi_B(0)$ in the matrix U_0 respectively.

By analogy with (7.2), we can write that

$$\bar{B}_0 = \bar{B}_1 V_0 ,$$

where V_0 is the low triangular matrix of the matrix $\bar{B}_0$ dimension:

$$V_0 = \begin{bmatrix} 1 & & & & -B_B^1(0) \\ & \cdot & & & \\ & & \cdot & & \\ & & & \cdot & \\ & & & & 1 \end{bmatrix} ,$$

where

$$B_1 = \begin{bmatrix} \hat{D}_{0B}(0) & & & & & & & & & \\ \hat{B}_{0B}(0) & -I & & & & & & & & \\ & G(1) & G(1)B_B^1(0) & D_B(1) & & & & & & \\ & A(1) & A(1)B_B^1(0) & B_B(1) & -I & & & & & \\ & & & & G(2) & D_B(2) & & & & \\ & & & & A(2) & B_B(2) & -I & & & \\ & & & & & & \ddots & & & \\ & & & & & & & G(T-1) & D_B(T-1) & \\ & & & & & & & A(T-1) & B_B(T-1) & -I \end{bmatrix}$$

The dimension and location of the matrix $-B_B^1(0)$ in V_0 coincide with the dimension and location of the matrix $B_B^1(0)$ in $\bar{B}_0$. The matrix $\bar{B}_1$ is obtained from $\bar{B}_0$ through interchange of submatrices

$$[-I:B_B^1(0)] \quad \text{and} \quad [-I:0] \quad ,$$

$$[G(1]:0:D_B(1)] \quad \text{and} \quad [G(1):G(1)B_B^1(0):D_B(1)] \quad ,$$

$$[A(1):0:B_B(1)] \quad \text{and} \quad [A(1):A(1)B_B^1(0):B_B(1)] \quad .$$

In accordance with Theorem 8.1, a matrix, obtained by cutting out the rows coinciding with the rows of submatrices $\hat{D}_{0B}(0)$ and $\hat{B}_{0B}(0)$and by cutting out the columns coinciding with the columns of submatrices $\hat{D}_{0B}(0)$ and $G(1)$, is nonsingular. Consequently the rows of the matrix

$$[G(1)B_B^1(0):D_B(1)]$$

are linearly independent, and by column interchange this matrix can reduce to the form

$$[G(1)B_B^1(0):D_B(1)] \quad = \quad [\hat{D}_{0B}(1):\hat{D}_{1B}(1)] \quad ,$$

where the matrix $\hat{D}_{0B}(1)$ is nonsingular and the matrix $\hat{D}_{1B}(1)$ is generated by columns $[G(1)B_B^1(0):D\ (1)]$, which are not in the matrix $\hat{D}_{0B}(1)$.

The matrices

$$[A(1)B_B^1(0):B_B(1)] \quad = \quad [\hat{B}_{0B}(1):\hat{B}_{1B}(1)]$$

and $\Phi_B(0)$ in matrix U_0, as well as the matrix $-B_B^{-1}(0)$ in the matrix V_0, are partitioned similarly.

Proceeding in a similar way, we obtain

$$\bar{B} \quad = \quad B^*V_{T-2}U_{T-2} \; \cdots\cdots \; V_0U_0 \quad = \quad \bar{B}U \quad , \qquad (7.3)$$

where

$$B^* \quad = \quad \begin{bmatrix} \hat{D}_{0B}(0) & & & & & & \\ \hat{B}_{0B}(0) & -I & & & & & \\ & G(1) & \hat{D}_{0B}(1) & & & & \\ & A(1) & \hat{B}_{0B}(1) & -I & & & \\ & & & & \ddots & & \\ & & & & & G(T-1)\hat{D}_{0B}(T-1) & \\ & & & & & A(T-1)\hat{B}_{0B}(T-1) & -I \end{bmatrix} .$$

The matrix $\hat{D}_{0B}(t)$ may include the columns of the matrix $D_B(t)$ and some columns of the matrix $D_B(\tau)$ $(\tau = 0,1,\ldots,t-1)$ which are recomputed at the step t in the factorization process.

Evidently the matrices $\hat{D}_{0B}(t)$ $(t = 0,1,\ldots,T-1)$ are obtained in such a way as to coincide with the local bases, which were defined in sections 3, 4.

Taking into account the interchange of basis columns in the factorization process, we can write the basic variables as

$$\{u_B,x\} \quad = \quad \{\hat{u}_{0B}(0),x(1),\hat{u}_{0B}(1),\ldots,\hat{u}_{0B}(T-1),x(T-1)\} \quad ,$$

where vector $\hat{u}_{0B}(t)$ corresponds to matrix $\hat{D}_{0B}(t)$ $(t = 0,1,\ldots,T-1)$.

Thus the matrices U_t and V_t $(t = 0,1,\ldots,T-2)$ are

$$U_t = \begin{bmatrix} 1 & & & & 0 & & \\ & 1 & & & & & \\ & & \ddots & 0 \ldots \Phi_0^{t+1}(t) \ldots 0 \ldots \Phi_i^j(t) \ldots 0 \ldots \Phi_t^{T-1}(t) \ldots 0 & & & \\ & & & \ddots & & 0 & \\ & & 0 & & \ddots & & \\ & & & & & 1 & \\ & & & & & & 1 \\ & & & & & & & 1 \end{bmatrix},$$

and

$$V_t = \begin{bmatrix} 1 & & & & 0 & & \\ & 1 & & & & & \\ & & \ddots & 0 \ldots -B_0^{t+1}(t) \ldots 0 \ldots -B_i^j(t) \ldots 0 \ldots -B_t^{T-1}(t) \ldots 0 & & & \\ & & & \ddots & & 0 & \\ & & 0 & & \ddots & & \\ & & & & & 1 & \\ & & & & & & 1 \\ & & & & & & & 1 \end{bmatrix}.$$

At each simplex iteration it is necessary to solve three systems of linear equations:

(1) determination of a basis solution;

(2) computation of coefficients {v,y}, which express vector

$$Y_j(t_1) \;=\; (0,\ldots,0,d_j^T(t_1),b_j^T(t_1),0,\ldots,0)^T$$

to be introduced into the basis;

(3) determination of the simplex-multipliers.

Now we describe briefly these procedures for factorized representation of the basis.

(1) Vector $X = (u_B,x)$ is determined from the solution of the system

$$\bar{B}X \;\equiv\; B^*UX \;=\; B^*V_{T-2}\ldots U_0X \;=\; b \quad , \qquad (7.4)$$

where b is the constraint vector of Problem 1.1.

Denote

$$X^* \;=\; UX \quad ,$$

then the determination of the vector X reduces to subsequent solution of two systems of linear equations in forward and backward transformations:

$$B^*X^* \;=\; b \quad , \qquad (7.5)$$

$$UX \;=\; X^* \quad . \qquad (7.6)$$

The solution of (7.5) is determined by recurrent formulas:

$$\begin{aligned} \hat{u}^*_{0B}(t) &= \hat{D}^{-1}_{0B}(t)\;\; f(t) - G(t)x^*(t)) \quad (t=0,\ldots,T-1) \quad , \\ x^*(t+1) &= A(t)x^*(t) + \hat{B}_{0B}(t)\hat{u}^*_{0B}(t) \quad (t=0,\ldots,T-1) \quad (7.7) \\ x^*(0) &= x(0) \quad . \end{aligned}$$

The system (7.6), considering (7.3), can be written as

$$X \;=\; U_0^{-1} \;\ldots\; V_{T-2}^{-1}X^* \quad .$$

It is easy to see that the matrices U_t^{-1} and V_t^{-1} are obtained from the matrices U_t and V_t by simply changing the signs of the elements which are above the main diagonal. Therefore the solution of the system (7.6) reduces to the recurrent formulas:

$$x(T) = x^*(T) ,$$

$$u(T-1) = u^*(T-1) ,$$

$$x(t) = x^*(t) + \sum_{i=0}^{t-1} \sum_{j=t}^{T-1} [B_i^j(t):0]u_{0B}(j) , \quad (t = T-1,\dots,1) \tag{7.8}$$

$$\hat{u}_{0B}(t) = \hat{u}_{0B}^*(t) - \sum_{i=0}^{t} \sum_{j=t+1}^{T-1} [\Phi_i^j(t):0]\hat{u}_{0B}(j) , \quad (t = T-2,\dots,0) .$$

Here the notations $[B_i^j(t):0]$ and $[\Phi_i^j(t):0]$ denote that the matrices $B_i^j(t)$ and $\Phi_i^j(t)$ are supplemented zeros if necessary for correctness of multiplying.

(2) The coefficients

$$\bar{Y}_j(t_1) = (\hat{v}_{0B}(0), y(1),\dots,y(T))$$

which express the vector $Y_j(t_1)$ by the basis are computed from the solution of the system

$$\bar{B}\bar{Y}_j(t_1) = Y_j(t_1) .$$

At direct run we can find vector (v^*,y^*):

$$\hat{v}_{0B}^*(t) = 0 ,$$

$$y^*(t+1) = 0 , \quad (t = 0,\dots,t_1-1) ,$$

$$\hat{v}_{0B}^*(t_1) = \hat{D}_{0B}^{-1}(t_1)d_j(t_1) ,$$

$$y^*(t_1+1) = \hat{B}_{0B}(t_1)\hat{v}_{0B}^*(t_1) - b_j(t_1) , \tag{7.9}$$

$$\hat{v}_{0B}^*(t) = -\hat{D}_{0B}^{-1}(t)G(t)y^*(t) ,$$

$$y^*(t+1) = A(t)y^*(t) + \hat{B}_{0B}(t)\hat{v}_{0B}^*(t) , \quad (t = t_1+1,\dots,T-1)$$

At reverse run we can find vector (v,y);

$$y(T) = y^*(T) \quad ,$$

$$\hat{v}_{0B}(T-1) = v^*_{0B}(T-1) \quad ,$$

$$y(t) = y^*(t) + \sum_{i=0}^{t-1} \sum_{j=t}^{T-1} [B_i^j(t):0]\hat{v}_{0B}(j) \quad , \quad (t = T-1,\dots,1) \quad , \qquad (7.10)$$

$$\hat{v}_{0B}(t) = \hat{v}^*_{0B}(t) - \sum_{i=0}^{t} \sum_{j=t+1}^{T-1} [\Phi_i^j(t):0]\hat{v}_{0B}(j) \quad , \quad (t = T-2,\dots,0) \quad .$$

(3) Determination of the simplex-multipliers $\{\lambda(0),p(1),\dots,\lambda(T-1),p(T)\}$ is carried out in a similar way by formulas

$$p(T) = a(T) \quad ,$$

$$\lambda(t) = p(t+1)\hat{B}_{0B}(t)\hat{D}_{0B}^{-1}(t), \quad (t = T-1,\dots,0) \quad , \qquad (7.11)$$

$$p(t) = p(t+1)A(t) - \lambda(t)G(t), \quad (t = T-1,\dots,1) \quad .$$

One can see that the formulas (7.7) to (7.10) are the explicit expression of Procedure 1 (sections 3,4) for determination of basic variables and coefficients, expressing a column not in the basis by the basis columns. The formulas (7.11) for determination of simplex-multipliers coincide with the formulas of Procedure 3 (section 5).

Thus the method of solution of Problem 1.1 which has been described in sections 3 to 6 is a realization of factorized representation of the inverse as applied to DLP problems.

We now describe the general procedure of the dynamic simplex-method.

8. GENERAL PROCEDURE OF DYNAMIC SIMPLEX-METHOD

Let at some iteration there be known:

$\{\hat{D}_{0B}^{-1}(t)\}$ - the inverses of local bases;

$\{u_{0B}(t)\}$ - the basic feasible control;

$\{x(t)\}$ - the corresponding trajectory;

$\{\lambda(t), p(t)\}$ - the dual variables (simplex-multipliers).

As in the ordinary static simplex-method, one can introduce artificial variables at zero iteration. In that case the zero iteration local bases are the identity matrices.

In accordance with Sections 3 to 7 the general procedure of the dynamic simplex method comprises the following stages:

1. Choose some pair of indices (j,t_1), for which

$$z_j(t_1) - c_j(t_1) =$$

$$= \lambda(t_1)d_j(t_1) - p(t_1+1)b_j(t_1) < 0, \quad (j,t_1) \in I_N(u) .$$

Usually a pair (j,t_1) with maximal absolute value of $z_j(t_1) - c_j(t_1)$ is selected.

If all

$$z_j(t) - c_j(t) \geq 0 , \qquad (j,t) \in I_N(u) \quad ,$$

then we have an optimal solution of the problem.

2. Define sequences of vectors v* and y* (forward transformation:

$$\hat{v}^*_{0B}(t) = 0$$

$$y^*(t+1) = 0$$

$$(t = 0,1,\ldots,t_1 - 1)$$

$$\hat{v}^*_{0B}(t_1) = \hat{D}_{0B}^{-1}(t_1)d_j(t_1)$$

$$y^*(t_1+1) = \hat{B}_{0B}(t_1)\hat{v}^*_{0B}(t_1) - b_j(t_1)$$

$$\hat{v}^*_{0B}(t) = -\hat{D}^{-1}_{0B}(t)G(t)y^*(t)$$
$$y^*(t+1) = A(t)y^*(t) + \hat{B}_{0B}(t)\hat{v}^*_{0B}(t)$$
$$(t = t_1+1,\ldots,T-1)$$

3. Define coefficients

$$\bar{Y}_j(t_1) = (v,y)$$

(backward transformation):

$$y(T) = y^*(T)$$
$$\hat{v}_{0B}(T-1) = \hat{v}^*_{0B}(T-1)$$
$$y(t) = y^*(t) + \sum_{j=t}^{T-1}\sum_{i=0}^{t-1}\left[B_i(t):0\right]\hat{v}_{0B}(j)$$
$$(t = T-1, T-2,\ldots,1)$$
$$\hat{v}_{0B}(t) = \hat{v}^*_{0B}(t) - \sum_{j=t+1}^{T-1}\sum_{i=0}^{t}\left[\Phi^j_i(t):0\right]\hat{v}_{0B}(j)$$
$$(t = T-2, T-3,\ldots,1,0)$$

4. Find the index t_2 of the column to be removed from the basis:

$$\theta_0 = \min_{\hat{v}_{0i}(t)>0} \frac{\hat{u}_{0i}(t)}{\hat{v}_{0i}(t)} = \frac{\hat{u}_{0\ell}(t_2)}{\hat{v}_{0\ell}(t_2)} .$$

If all $\hat{v}_{0i}(t) \leq 0$, then the solution is unbounded.

5. Compute the new basic feasible control $\{u'(t)\}$:

$$u'_i(\tau) = \begin{cases} u_i(\tau) - \theta_0 v_{iB}(\tau) & (i,\tau) \in I_B(u) \\ \theta & (i,\tau) = (j,t_1) \\ 0 & (i,\tau) \in I_N(u),\ (i,\tau) \neq (j,t_1) \end{cases}$$

6. Transform the local bases:

a) set $t = t_2$;

b) if $t \geq t_1$, then go to stage e);

c) choose the nonzero element in the pivot row of the matrix $\Phi_B(t)$. (The index of the pivot row equals the index of the column to be removed from the local basis $\hat{D}_{0B}(t)$.)

d) let the pivot element of the matrix $\Phi_B(t)$ correspond to the component of the basic control, which was recomputed into the local basis at step $t + \tau$. Then:

- interchange the positions between local bases $\hat{D}_{0B}(t)$ and $\hat{D}_{0B}(t + \tau)$
- set $t \to t + \tau$
- go to stage b);

e) if $t = T - 1$, then go to stage f;

f) replace the column to be removed by the column to be introduced into $\hat{D}_{0B}(t)$.

7. Compute the dual variables

$$p(T) = a(T)$$

$$\lambda(t) = p(t+1)\hat{B}_{0B}(t)\hat{D}_{0B}^{-1}(t) \qquad (t = T-1, \ldots, 1, 0)$$

$$p(t) = p(t+1)A(t) - \lambda(t)G(t) \qquad (t = T-1, \ldots, 1) .$$

Go to stage 1.

It should be noted that only the general scheme of the algorithm is given here. The concrete realization of the algorithm depends on the specific of a problem, type of computer, ways of selecting the column to be introduced into basis, etc.

It should also be noted that for realization of the algorithm it is sufficient to operate only with matrices

$$\hat{D}_{0B}^{-1}(t);\ \Phi_B(t),\ \hat{B}_{0B}(t),\ B_B^1(t),\ G(t),\ A(t) \qquad (t = 0, 1, \ldots, T-1) .$$

9. DEGENERACY

It was assumed above that all basic feasible controls were nondegenerate.

This assumption was necessary in order to guarantee that for each successive set of local feasible bases the associated value of the objective function is larger than those that precede it. Hence we will reach the optimal solution in a finite number of possible sets of local feasible bases.

For the degeneracy case there is the possibility of computing θ_0 at step 4 of section 8, for which $\theta_0 = 0$. Therefore the choice of a vector to be removed from and a vector to be introduced into the set of local bases will give a new basic feasible control with the value of the objective function being equal to the preceding one. Thus cycling of the procedure is possible.

Therefore a special rule for selecting the column to be removed should be elaborated to overcome cycling in the case of degeneracy.

For that we can use the method of overcoming degeneracy of the revised simplex-method [1,14].

We can use this method for the dynamic simplex-method as well if the columns of the inverse $\bar{B}^{-1}$ (see (7.1)) can be computed.

The j-th column y_j of the inverse $\bar{B}^{-1}$ is a solution of the system of equations:

$$\bar{B}y_j = e_j \quad , \tag{9.1}$$

where e_j is the unit vector of dimension $(m+n)T$ with the j-th component equal to one.

The system (9.1) can be solved by using the factorized representation of the basis matrix, which is similar to the solution of system (7.4).

10. EVALUATION OF THE ALGORITHM

In this section we give some theoretical evaluation of the dynamic simplex-method.

As was noted in section 8, for realization of the algorithm it is sufficient to operate only with the matrices $\hat{D}_{0B}^{-1}(t)$; $\Phi_B(t)$, $\hat{B}_{0B}(t)$, $B_B^1(t)$, $G(t)$, $A(t)$ $(t=0,1,\dots,T-1)$.

Theorem 10.1 *The number of columns of matrices* $\Phi_B(t)$ *and* $B_B^1(t)$ *does not exceed* n.

Proof Let 2t steps of the factorization process be carried out.

Then the formula (7.3) can be rewritten as

$$\bar{B} = B_{2t-1} V_{t-1} U_{t-1} \cdots V_0 U_0 .$$

On the main diagonal of the matrix $\bar{B}_{2t-1}$ there is the submatrix

$$F = \begin{bmatrix} \hat{D}_{0B}(t) & \hat{D}_{1B}(t) \\ \hat{B}_{0B}(t) & \hat{B}_{1B}(t) \end{bmatrix} .$$

The columns of the submatrix F are linearly independent, as the matrix B_{2t-1} is nonsingular. Consequently the number of columns of matrices $\hat{D}_{1B}(t)$ and $\hat{B}_{1B}(t)$ cannot be larger than n. Hence one can obtain the statement of the theorem.

The matrices $\hat{D}_{0B}^{-1}(t)$, $\hat{B}_{0B}(t)$, $G(t)$, $A(t)$ have dimensions $(m \times m)$, $(n \times m)$, $(m \times n)$, $(n \times n)$ respectively.

Hence the algorithm operates only with the set of T matrices, each containing no more than m or n columns.

At the same time the straightforward application of the simplex-method to Problem 1.1 (in the space of (u,x)) leads to the necessity of operating with the basis matrix of dimension

$(m+n)T \times (m+n)T$ or of dimension $mT \times mT$, if the state variables are excluded beforehand.

Thus in some respects the dynamic simplex-method realizes a decomposition of the problem that allows a substantial saving in the number of arithmetical operations and in the core memory.

As was mentioned above, the DLP Problem 1.1 can be considered as some "large" static LP problem and thus the revised simplex-method can be used for its solution.

Let us compare the effectiveness of the dynamic simplex-method and the revised simplex-method as applied to Problem 1.1.

It is known that at each iteration the revised simplex-method requires of the order of k^2 multiplications for transformation of the inverse, where k is the number of rows of the basic matrix. Hence the total number of multiplications for transformation of the basis is of order $(m+n)^2T^2$. To compute the coefficients which express the column to be introduced into a basis in terms of columns of the current basis, the revised simplex-method requires some $(m+n)^2T$ multiplications.

Now we shall evaluate the number of multiplications for the dynamic simplex-method. It was shown that at one interchange the local bases are transformed by multiplication on the elementary column or row matrix. The interchange of columns between two neighbouring local bases $\hat{D}_{0B}(t)$ and $\hat{D}_{0B}(t+1)$ requires no more than $3(m+n)^2$ multiplications. (The matrices $\hat{D}_{0B}^{-1}(t)$, $\hat{B}_{0B}(t)$, $\Phi_B(t)$, $B_B^1(t)$, $\hat{D}_{0B}^{-1}(t+1)$, $\hat{B}_{0B}(t+1)$, $\Phi_B(t+1)$ are transformed.) In the worst case, when the column to be removed from the basis is transformed from the local bases $\hat{D}_{0B}(0)$ into the local basis $\hat{D}_{0B}(T-1)$, one needs T interchanges. We assume that the average number of interchanges is T/2. Thus the dynamic simplex-method requires approximately $1.5(m+n)^2T$ multiplications for transformation of local bases for one iteration.

Computation of the coefficients expressing vector to be introduced into a basis requires about $(m+n)^2T$ multiplications. It is important that only part of local bases are transformed at each iteration. In addition local bases can be represented in factorized form, thus enabling use of the effective procedures of static LP [1,14].

Solution of Problem 1.1 by the revised simplex-method requires storage of the inverse of dimension $(m+n)T \times (m+n)T$. The dynamic simplex-method requires storage of only T matrices of dimension $m \times m$ $(\hat{D}_{0B}^{-1}(t), \hat{B}_{0B}(t))$ and $T-1$ matrices of dimension $m \times n$ $(\Phi_B(t))$ and $n \times n$ $(B_B^1(t))$.

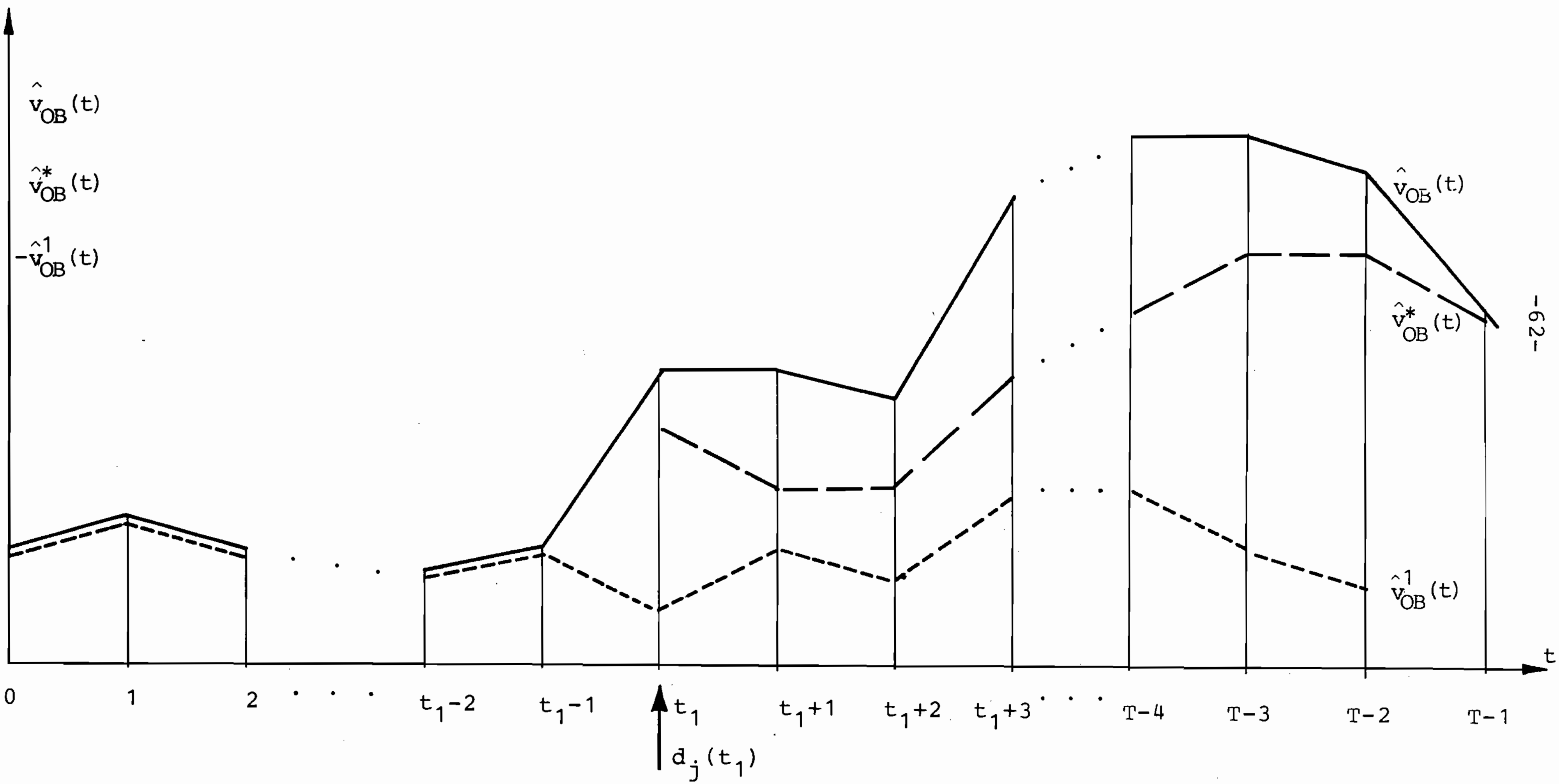
Thus comparing the estimates for the static and dynamic algorithms for solution of Problem 1.1, one can see that the volume of computation and the core memory increases linearly with T for the dynamic algorithm and by quadratic law for the static algorithm.

CONCLUSION

The general scheme and basic theoretical properties of the dynamic simplex-method specially developed for solution of dynamic linear programs is described and discussed.

Theoretical reasonings show that this algorithm may serve as a base for developing effective computer codes for the solution of DLP problems. However the final judgment of the efficiency of the algorithm can be made only after definite period of its exploitation in practice.

It should also be very interesting to compare (both from the theoretical and the computational point of view) the approach given in this paper with the finite-step DLP algorithm based on the Dantzig-Wolfe decomposition principle [18] and other methods of solving DLP problems [7,8,9,16].



$\hat{v}_{OB}(t)$
$\hat{v}^{*}_{OB}(t)$
$-\hat{v}^{1}_{OB}(t)$
$\hat{v}_{OB}(t)$
$\hat{v}^{*}_{OB}(t)$
$\hat{v}^{1}_{OB}(t)$
t
0
1
2
t_1-2
t_1-1
t_1
t_1+1
t_1+2
t_1+3
T-4
T-3
T-2
T-1
$d_j(t_1)$

Fig. 1

Variables											
u(0)	x(1)	...	x(t)	u(t)	x(t+1)	...	x(T-1)	u(T-1)	x(T)	Constraints	Right-hand side constants
-B(0)	I									=	$A(0)x^0$
D(0)		⋱								=	$f(0) - G(0)x^0$
											⋮
			-A(t)	-B(t)	I					=	0
			G(t)	D(t)		⋱				=	f(t)
											⋮
							-A(T-1)	-B(T-1)	I	=	0
							G(T-1)	D(T-1)		=	f(T-1)
Performance Index Constants										Max	
0				...				0	a(T)		

Table 1. Staircase Control Structure

Variables								
u(0)	u(1)	...	u(t-1)	u(t)	...	u(T-1)	Constraints	Right-hand Side Constants
D(0)							=	h(0)
W(1,0)	D(1)						=	h(1)
. . .	. . .	. .	. . .	. . .	. .	. . .	. . . .	. . . . . .
W(t,0)	W(t,1)	...	W(t,t-1)	D(t)			=	h(t)
. . .	. . .	. .	. . .	. . .	. .	. . .	. . . .	. . . . . .
W(T-1,0)	W(T-1,1)	...	W(T-1,t-1)	W(T-1,t)	...	D(T-1)	=	h(T-1)
Performance Index Constants							Max	
c(0)	c(1)	...	c(t-1)	c(t)	...	c(T-1)		

Table 2. Block Triangular LP Structure

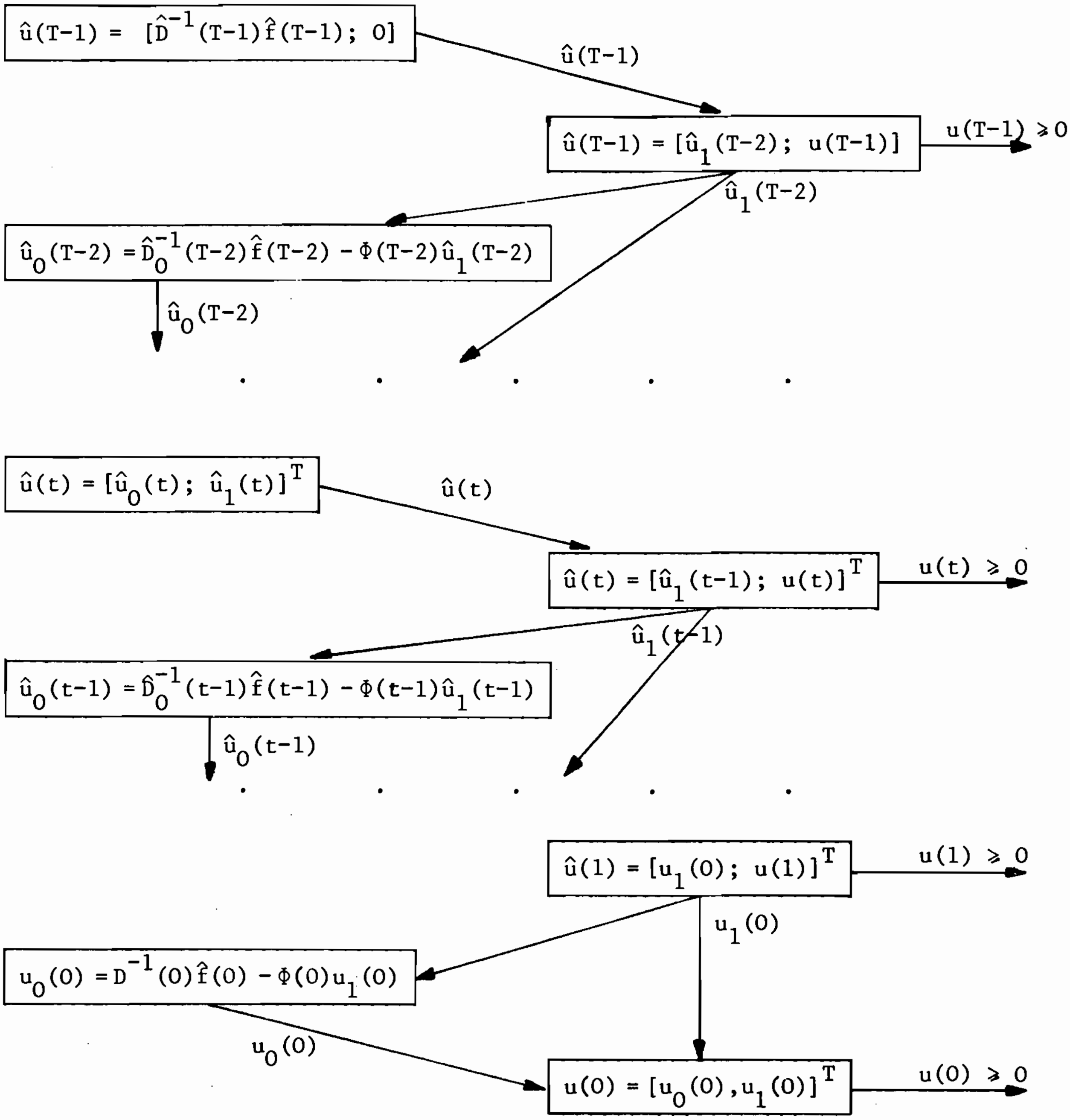


Table 3. Procedure of basic feasible control computation for a given set of local bases $\{D_0(t)\}$

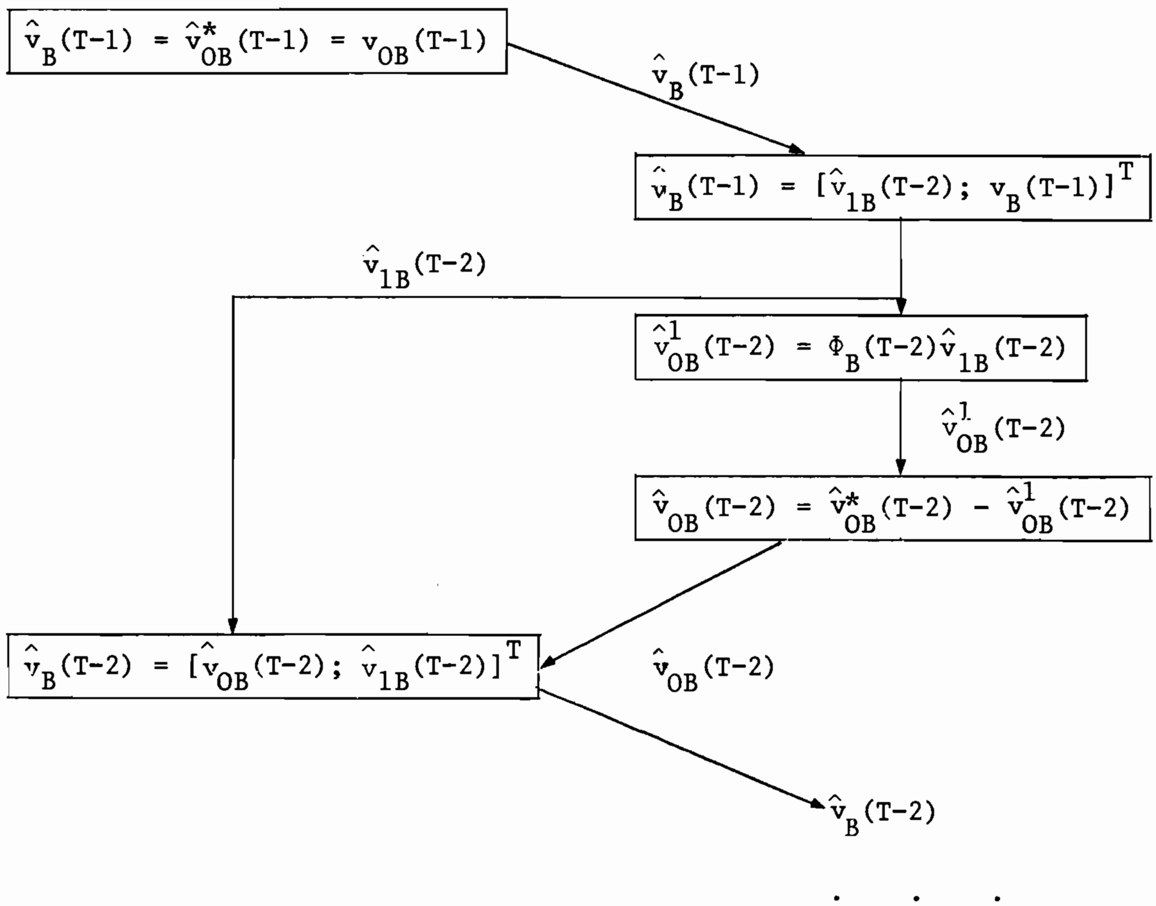


Table 4. Procedure of the computation of the basic feasible control variation.

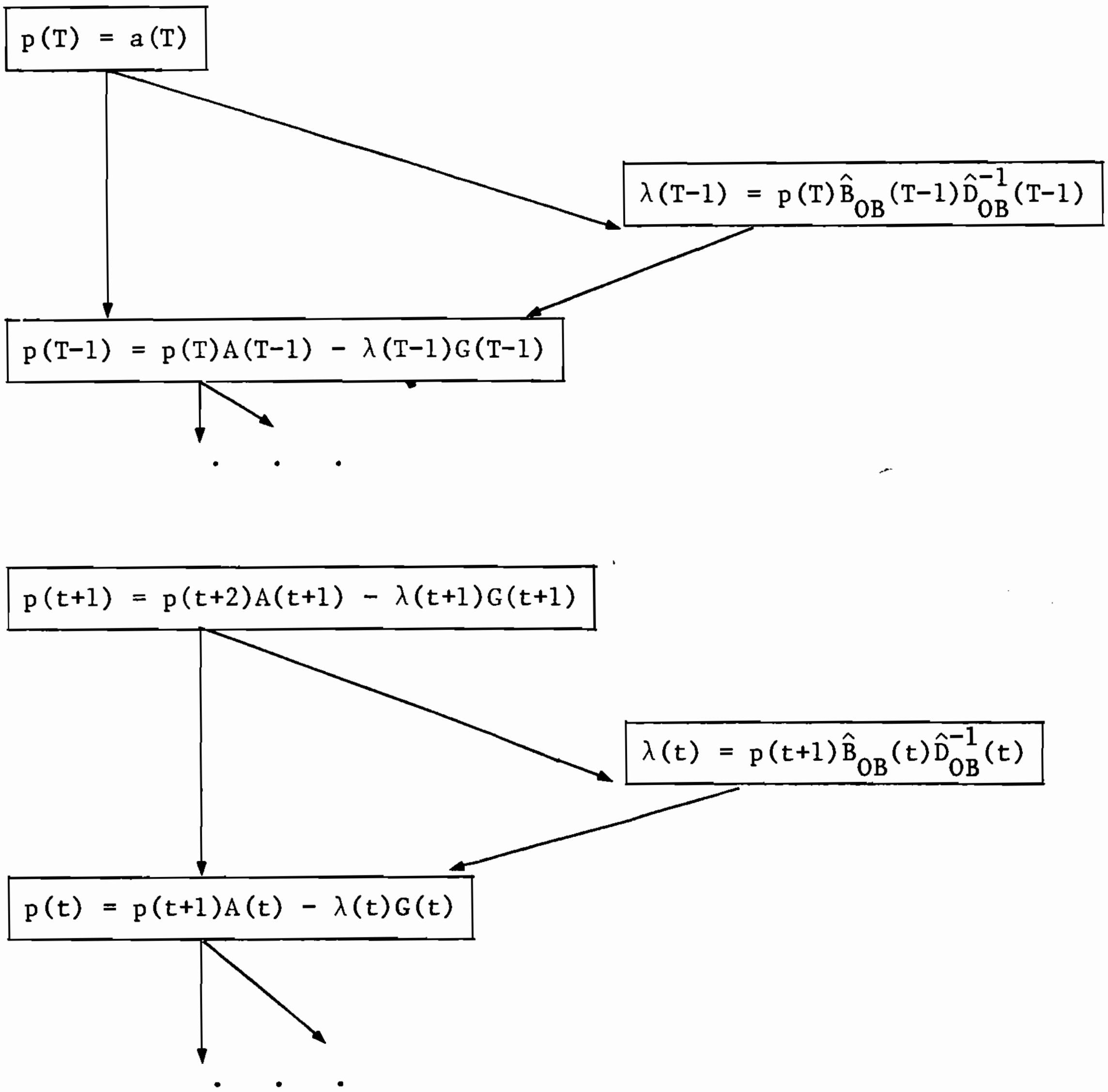


Table 5. Procedure of the simplex-multipliers computation.

References

[1] Dantzig, G.B., *Linear Programming and Extensions*, University Press, Princeton, N.J., 1963.

[2] Kantorovitch, L.V., *The Best Use of Economic Resources*, Harvard University Press, Cambridge, Mass., 1965.

[3] Propoi, A.I., *Problems of Dynamic Linear Programming*, RM-76-78, International Institute for Applied Systems Analysis, Laxenburg, Austria, 1976.

[4] Haefele, W. and A.S. Manne, *Strategies of a Transition from Fossil to Nuclear Fuels*, RR-74-07, International Institute for Applied Systems Analysis, Laxenburg, Austria, 1974.

[5] Aganbegian, A.G. and K.K. Valtukh, Izpol'zovanie Narodnokhoziastvennykh Modelei v Planirovanii (Utilization of National Economy Models in Planning), *Ekonomica*, M., 1975 (in Russian).

[6] Swart, W. a.o. Expansion planning for a large dairy farm, in H. Salkin and J. Saha, eds., *Studies in Linear Programming*, North-Holland, N.Y., 1975.

[7] Glassey, C.R., Dynamic Linear Programming for Production Scheduling, *Operations Research*, 18 1(1970).

[8] Ho, J.K. and A.S. Manne, Nested Decomposition for Dynamic Models, *Mathematical Programming*, 6 2(1974).

[9] Propoi, A.I. and A.B. Yadykin, Parametric Iterative Methods for Dynamic Linear Programming. I. Non-degenerative Case, *Avtomatika i Telemekhanika*, 12, (1975); II. General Case, *Avtomatika i Telemekhanika*, 1, (1976) (in Russian).

[10] Krivonozhko, V.E. and A.I. Propoi, A method for DLP with the use of basis matrices factorization, *IX Internat. Symp. on Mathematical Programming, Budapest, 1976*

[11] Krivonozhko, V.E. and A.I. Propoi, Method of successive plan improvement for DLP, *Izv. Akad. Nauk SSSR, Technicheskaia Kibernetika*, N2, 1977 (in Russian).

[12] Propoi, A.I., *Dual Systems of DLP*, RR-77-9 , International Institute for Applied Systems Analysis, Laxenburg, Austria, 1977.

[13] Propoi, A., Elementy Teorii Optimalnykh Discretnykh Protsessov (Elements of the Theory of Optimal Discrete Systems), *Nauka*, Moscow, 1973 (in Russian).

[14] Gass, S.L., *Linear Programming: Methods and Applications*, McGraw-Hill Co., 1975.

[15] Gantmacher, F.R., *The Theory of Matrices*, Chelsea Publishing Co., N.Y., 1960.

[16] Krivonozhko, V.E. and S.P. Chebotarev, On the Use of Basis Factorization for Dynamic Linear Program Solving, *Avtomatika i Telemekhanika*, 7, (1976), (in Russian).

[17] Forrest, J.J.H. and J.A. Tomlin, Updating triangular factors of the basis to maintain sparsity in the product form simplex method, *Math. Programming*, N2, (1972).

[18] Krivonozhko, V.E., Decomposition Method for Dynamic Linear Programming, *Izv. Akad. Nauk SSSR, Tekhnicheskaia Kibernetika*, 6 (1976) (in Russian).